Unable to locate subtitle

Framework AWS Well-Architected

Framework AWS Well-Architected: ***Unable to locate subtitle***

Table of Contents

Framework AWS Well-Architected

Data di pubblicazione: 27 giugno 2024 (Revisioni del documento)

Il Framework AWS Well-Architected aiuta a comprendere i pro e i contro delle decisioni prese durante la progettazione di sistemi in AWS. Utilizzando il Framework, scoprirai le best practice architetturali per progettare e gestire sistemi affidabili, sicuri, efficienti, convenienti e sostenibili nel cloud.

Introduzione

Il Framework AWS Well-Architected aiuta a comprendere i pro e i contro delle decisioni prese durante la progettazione di sistemi in AWS. Utilizzando il Framework, scoprirai le best practice architetturali per progettare e gestire carichi di lavoro affidabili, sicuri, efficienti, convenienti e sostenibili nel Cloud AWS. Permette di misurare in modo coerente le architetture rispetto alle best practice e identificare le aree da migliorare. Il processo di revisione di un'architettura consiste in una conversazione costruttiva sulle decisioni relative all'architettura e non è un meccanismo di audit. Disporre di sistemi ben architettati aumenta notevolmente la probabilità di successo aziendale.

AWS Solutions Architects vanta anni di esperienza nell'architettura di soluzioni in un'ampia gamma di business e casi di utilizzo. Abbiamo supportato migliaia di clienti nella progettazione e revisione delle loro architetture su AWS. Grazie a questa esperienza, abbiamo identificato best practice e strategie principali per i sistemi di architettura nel cloud.

Il Framework AWS Well-Architected documenta un insieme di domande fondamentali per capire se un'architettura specifica si allinea bene con le best practice del cloud. Il canone fornisce un approccio coerente per la valutazione dei sistemi rispetto alle qualità che ti aspetti da sistemi basati sul cloud moderni e i rimedi necessari per raggiungere tali qualità. Man mano che AWS continua a evolversi e noi continuiamo a imparare di più dal lavoro che svolgiamo con i nostri clienti, continueremo a ridefinire la definizione di canone di architettura.

Questo canone è rivolto a chi svolge ruoli tecnologici, ad esempio ai Chief Technology Officer (CTO), ai progettisti, agli sviluppatori e ai membri dei team operativi. Descrive le best practice e le strategie AWS da usare per la progettazione e il funzionamento di un carico di lavoro cloud, e fornisce collegamenti a ulteriori dettagli di implementazione e pattern architetturali. Per ulteriori informazioni, consulta la Homepage del Canone di architettura AWS.

AWS offre anche un servizio gratuito di revisione dei carichi di lavoro. Il AWS Well-Architected Tool (AWS WA Tool) è un servizio cloud che fornisce un approccio coerente per la revisione

e la valutazione della tua architettura secondo il Framework AWS Well-Architected. AWS WA Tool fornisce raccomandazioni per rendere i tuoi carichi di lavoro più affidabili, sicuri, efficienti e convenienti.

Per aiutarti ad applicare le best practice, abbiamo creato AWS Well-Architected Labs, che fornisce un repository di codice e documentazione per un'esperienza concreta di implementazione delle best practice. Abbiamo anche collaborato con partner APN (AWS Partner Network) selezionati, che sono membri del Programma Partner AWS Well-Architected. Tali partner AWS vantano una conoscenza approfondita di AWS e possono aiutarti nella revisione e nel miglioramento dei tuoi carichi di lavoro.

Definizioni

Tutti i giorni, gli esperti AWS supportano i clienti nella progettazione di sistemi di architettura per sfruttare le best practice nel cloud. Ti aiutiamo a trovare i compromessi relativi all'architettura nel processo di evoluzione dei tuoi progetti. Quando distribuisci questi sistemi in ambienti live, analizziamo le prestazioni di questi sistemi e le conseguenze dei suddetti compromessi.

Sulla base di quello che abbiamo imparato, abbiamo creato il Framework AWS Well-Architected, che fornisce a clienti e partner un insieme coerente di best practice per valutare le architetture, e comprende un insieme di domande che puoi utilizzare per valutare se la tua architettura è ben allineata alle best practice AWS.

Il Framework AWS Well-Architected si basa su sei pilastri: eccellenza operativa, sicurezza, affidabilità, efficienza delle prestazioni, ottimizzazione dei costi e sostenibilità.

Tabella 1. I pilastri del Framework AWS Well-Architected

Nome	Descrizione
Eccellenza operativa	Comprende la capacità di supportare lo sviluppo ed eseguire carichi di lavoro in modo efficace, ottenere informazioni approfondite sulle loro operazioni e migliorare continuam ente i processi e le procedure di supporto per offrire valore aggiunto.
Sicurezza	Il pilastro della sicurezza descrive come sfruttare le tecnologie cloud per proteggere

Nome	Descrizione
	dati, sistemi e asset in modo da migliorare la sicurezza.
Affidabilità	Il principio dell'affidabilità comprende la capacità di un carico di lavoro di eseguire la funzione attesa in modo corretto e coerente quando previsto. Include la possibilità di utilizzare e testare il carico di lavoro per tutto il ciclo di vita. Questo documento fornisce linee guida dettagliate sulle best practice per l'implementazione di carichi di lavoro affidabili in AWS.
Efficienza delle prestazioni	L'abilità di utilizzare in modo efficiente le risorse di elaborazione per soddisfare i requisiti del sistema e conservare tale efficienza a seconda dei cambiamenti della domanda e dell'evol uzione delle tecnologie.
Ottimizzazione dei costi	La capacità di eseguire sistemi per fornire valore aziendale al minor prezzo possibile.
Sostenibilità	La capacità di migliorare continuamente l'impatto sulla sostenibilità riducendo il consumo energetico e aumentando l'efficie nza di tutti i componenti di un carico di lavoro, massimizzando i benefici delle risorse di cui è stato eseguito il provisioning e riducendo al minimo le risorse totali richieste.

Nel Framework AWS Well-Architected, si utilizzano i seguenti termini:

- Un componente è un codice, una configurazione e delle risorse AWS che insieme riescono a soddisfare un requisito di un carico di lavoro. Spesso un componente è l'unità di proprietà tecnica ed è disaccoppiato da altri componenti.

- Con il termine carico di lavoro ci riferiamo all'insieme di componenti che forniscono valore aziendale. Un carico di lavoro, normalmente, è il livello di dettaglio comunicato dai leader aziendali e della tecnologia.
- Pensiamo a un' architettura come al modo in cui in componenti operano insieme in un carico di lavoro. Il modo di comunicare e di interagire dei componenti è spesso l'aspetto principale dei diagrammi architetturali.
- Le tappe fondamentali indicano cambiamenti chiave della tua architettura man mano che si evolve nel corso del ciclo di vita del prodotto (progettazione, test, messa online e produzione).
- Nell'ambito di un'organizzazione il portfolio delle tecnologie rappresenta l'insieme di carichi di lavoro necessari affinché l'azienda possa essere operativa.
- Il livello di impegno è la categorizzazione della quantità di tempo, sforzo e complessità che un'attività richiede per la sua realizzazione. Ogni organizzazione deve considerare le dimensioni e le competenze del team e la complessità del carico di lavoro per ottenere un contesto aggiuntivo che consenta di classificare correttamente il livello di impegno.
  - Alto: Il lavoro potrebbe richiedere più settimane o più mesi. Potrebbe essere suddiviso in molteplici fasi, rilasci e attività.
  - Medio: Il lavoro potrebbe richiedere più giorni o settimane. Potrebbe essere suddiviso in molteplici rilasci e attività.
  - Basso: Il lavoro potrebbe richiedere più ore o giorni. Potrebbe essere suddiviso in molteplici attività.

Quando progetti l'architettura dei carichi di lavoro, devi trovare dei compromessi tra i pilastri su cui si regge il tuo contesto aziendale. Questo tipo di decisioni aziendali deve essere alla base delle tue priorità ingegneristiche. Potresti ottimizzare per migliorare la sostenibilità e ridurre i costi a spese dell'affidabilità in ambienti di sviluppo oppure, per quanto riguarda le soluzioni mission-critical, potresti ottimizzare l'affidabilità a fronte di costi più elevati e di un impatto ambientale maggiore. Nelle soluzioni di e-commerce, le prestazioni possono avere un impatto sui profitti e sulla propensione all'acquisto da parte dei clienti. L'eccellenza in ambito di sicurezza e operatività generalmente non viene sacrificata rispetto agli altri pilastri.

Architettura

Negli ambienti on-premise, i clienti spesso hanno un team centrale per l'architettura delle tecnologie che funziona da livello superiore per altri team di prodotto o funzionalità, al fine di garantire che i team rispettino le best practice. I team dell'architettura delle tecnologie spesso sono composti da diversi

ruoli come il Technical Architect (infrastruttura), il Solutions Architect (software), il Data Architect, il Networking Architect e il Security Architect. Spesso i team usano TOGAF o framework di Zachman come parte delle competenze architetturali aziendali.

Noi di AWS preferiamo distribuire le competenze tra i team, invece di centralizzarle in un unico team. Quando si sceglie di distribuire il potere decisionale si corrono dei rischi, ad esempio il rischio di garantire che i team interni rispettino gli standard. Noi mitighiamo questo rischi in due modi. Innanzitutto, abbiamo le pratiche (Modalità per eseguire attività, processi, standard e norme accettate) che hanno lo scopo di permettere a ogni team di possedere tali competenze e ci serviamo di esperti che verificano che i team adottino standard più severi di quelli che devono rispettare. In secondo luogo, implementiamo meccanismi che eseguono controlli automatizzati per verificare che gli standard vengano rispettati.

> "Le buone intenzioni non bastano mai, per avere successo servono buoni meccanismi", Jeff Bezos.

Questo significa sostituire gli sforzi di una persona con meccanismi (spesso automatizzati) che verificano la conformità alle regole e ai processi. L'approccio distribuito è supportato dai principi di leadership di Amazone stabilisce una cultura tra tutti i ruoli che lavora a ritroso dal cliente. Il lavoro a ritroso è una parte fondamentale del nostro processo di innovazione. Partiamo dal cliente e da quello che vuole e sulla base di questo definiamo e indirizziamo i nostri sforzi. I team che mettono il cliente al centro sviluppano prodotti sulla base delle necessità del cliente.

Per l'architettura questo significa che ci aspettiamo che ogni team sia in grado di creare architetture e di seguire le best practice. Per aiutare i nuovi team ad acquisire queste competenze o i team esistenti ad alzare il livello, abilitiamo l'accesso a una community virtuale di ingegneri responsabili che possono eseguire la revisione dei loro progetti e aiutarli a comprendere le best practice di AWS. La community di ingegneri responsabili lavora per rendere visibili e accessibili le best practice. Uno dei modi per fare ciò, ad esempio, è servirsi delle lunchtime talk che si concentrano sull'applicazione di best practice a esempi reali. Le lunchtime talk sono registrate e possono essere utilizzate come materiale di onboarding per i nuovi membri del team.

Le best practice AWS sono il risultato della nostra esperienza nell'esecuzione di migliaia di sistemi su Internet. Preferiamo utilizzare i dati per definire le best practice, ma ci serviamo anche di esperti in materia, come i capo ingegneri. Quando i capo ingegneri vedono emergere nuove best practice, lavorano con la community per verificare che i team le rispettino. Con il tempo, queste best practice

vengono formalizzate nei nostri processi di revisione interna e nei meccanismi che rafforzano la compliance. Il Canone di architettura è l'implementazione del nostro processo di revisione interno rivolta ai clienti, in cui abbiamo codificato la nostra idea di ingegneria responsabile attraverso ruoli di campo come Solutions Architect e i team di ingegneria interni. Il canone di architettura è un meccanismo scalabile che consente di trarre vantaggio da questi insegnamenti.

Seguendo l'approccio della community di ingegneri responsabili con la proprietà distribuita dell'architettura, riteniamo che si possa ottenere un'architettura aziendale Well-Architected che si basa sulle necessità del cliente. I leader della tecnologia (come i CTO o i manager dello sviluppo) che eseguono revisioni Well-Architected tra tutti i carichi di lavoro ti permettono di comprendere più a fondo i rischi relativi al portfolio delle tecnologie. Tramite questo approccio puoi identificare dei temi tra i team che la tua organizzazione può affrontare tramite meccanismi, formazione o dialoghi informali in cui i capo ingegneri possono condividere le loro idee su aree specifiche con diversi team.

Principi generali di progettazione

Il Canone di architettura identifica una serie di principi generali per facilitare la corretta progettazione nel cloud:

- Smetti di ipotizzare quali siano le tue esigenze di capacità: quando prendi decisioni relative alla capacità prima della distribuzione di un sistema, potresti ritrovarti con risorse inattive o ad affrontare le conseguenze della capacità limitata. Con il cloud computing, questi problemi vengono risolti. Puoi utilizzare la capacità di cui hai bisogno e ridimensionare il sistema automaticamente.
- Esegui test dei sistemi su scala produttiva: nel cloud, puoi creare un ambiente di test su scala produttiva on demand, completare i test e ritirare le risorse. Poiché paghi per l'ambiente di test solo quando è in esecuzione, puoi simulare un ambiente live a un costo notevolmente inferiore rispetti ai test in locale.
- Automatizza pensando alla sperimentazione architettonica: l'automazione ti permette di creare e replicare i tuoi carichi di lavoro a basso costo e di evitare le spese della gestione manuale. Puoi tenere traccia delle modifiche all'automazione, effettuare l'audit dell'impatto e tornare ai parametri precedenti, se necessario.
- Considera le architetture evoluzionistiche.: in un ambiente tradizionale, le decisioni relative all'architettura spesso sono implementate come eventi singoli e statici, con poche versioni principali di un sistema durante il ciclo di vita. Alla luce del continuo cambiamento di un'azienda e del suo contesto, le decisioni iniziali potrebbero ostacolare la capacità del sistema di soddisfare i requisiti aziendali in evoluzione. All'interno del cloud, la capacità di automatizzare e testare on demand diminuisce il rischio di impatto dovuto alle modifiche della progettazione. Questo permette ai

sistemi di evolversi nel tempo, in modo che le aziende possano trarre vantaggio dalle innovazioni come pratica standard.

- Promuovi le architetture servendoti dei dati: nel cloud puoi raccogliere dati relativi all'impatto delle tue scelte architetturali sul comportamento del tuo carico di lavoro. Questo ti permette di prendere decisioni basate sui fatti su come migliorare il carico di lavoro. La tua infrastruttura cloud è un codice, quindi, puoi usare tali dati a vantaggio delle scelte e dei miglioramenti relativi all'architettura nel tempo.
- Migliora con le giornate di gioco: testa le prestazioni dell'architettura e dei processi pianificando regolarmente game day per simulare eventi della produzione. Questi ti aiuta a capire dove puoi apportare dei miglioramenti e ti può aiutare a sviluppare un'esperienza organizzativa nella gestione degli eventi.

I pilastri del framework

La creazione di un sistema software è molto simile alla costruzione di un edificio. Se le fondamenta non sono solide, possono emergere problemi strutturali che minano l'integrità e la funzionalità dell'edificio. Se nella creazione dell'architettura per soluzioni tecnologiche trascuri i sei principi di eccellenza operativa, sicurezza, affidabilità, efficienza delle prestazioni, ottimizzazione dei costi e sostenibilità, può diventare complicato sviluppare un sistema che soddisfi le tue aspettative e i tuoi requisiti. L'aggiunta di questi pilastri alla tua architettura ti aiuterà a produrre sistemi efficienti e stabili. Questo ti permetterà di concentrarti su altri aspetti della progettazione, come i requisiti funzionali.

Pilastri

- Eccellenza operativa
- Sicurezza
- Affidabilità
- Efficienza delle prestazioni
- Ottimizzazione dei costi
- Sostenibilità

Eccellenza operativa

Il principio dell'eccellenza operativa comprende la capacità di supportare lo sviluppo ed eseguire carichi di lavoro in modo efficace, ottenere informazioni approfondite sulle loro operazioni e migliorare continuamente i processi e le procedure di supporto per offrire valore aggiunto.

Il principio dell'eccellenza operativa offre una panoramica dei principi di progettazione, delle best practice e delle domande. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul principio dell'eccellenza operativa.

Argomenti

- Principi di progettazione
- Definizione
- Best practice
- Risorse

Principi di progettazione

Ecco i principi di progettazione per l'eccellenza operativa nel cloud:

- Organizza i team sulla base dei risultati aziendali: la capacità di un team di conseguire i risultati aziendali deriva dalla visione della leadership, dalle operazioni efficaci e da un modello operativo allineato all'azienda. È necessario che la leadership sia totalmente coinvolta e impegnata nella trasformazione delle operazioni nel cloud (CloudOps) con un modello operativo cloud adeguato che incentivi i team a operare nel modo più efficiente per raggiungere i risultati aziendali. Il modello operativo corretto include persone, processi e capacità tecnologiche per scalare, ottimizzare la produttività e favorire la differenziazione tramite l'agilità, la reattività e l'adattamento. La visione a lungo termine dell'organizzazione si traduce in obiettivi che vengono comunicati alle parti interessate dell'azienda e agli utenti dei tuoi servizi cloud. Gli obiettivi e i KPI operativi sono allineati a tutti i livelli. Questa procedura promuove il valore a lungo termine derivante dall'implementazione dei seguenti principi di progettazione.
- Implementa l'osservabilità per ottenere informazioni utili: acquisisci informazioni dettagliate su comportamento, prestazioni, affidabilità, costi e stato del carico di lavoro. Stabilisci indicatori chiave delle prestazioni (KPI) e usa la telemetria dell'osservabilità per prendere decisioni informate e agire tempestivamente quando i risultati aziendali sono a rischio. Migliora in modo proattivo le prestazioni, l'affidabilità e i costi sulla base di dati osservabili utilizzabili.
- Automatizza in sicurezza laddove possibile: nel cloud è possibile applicare a tutto il tuo ambiente la medesima disciplina di progettazione che utilizzi per il codice dell'applicazione. Definisci l'intero carico di lavoro e le relative operazioni (applicazioni, infrastruttura, configurazione e procedure) come codice e aggiornarlo. Quindi, automatizza le operazioni del carico di lavoro avviandole in risposta agli eventi. Nel cloud, utilizzi la sicurezza dell'automazione configurando i guardrail, tra cui il controllo della frequenza, le soglie di errore e le approvazioni. Un'automazione efficiente offre risposte coerenti agli eventi, limita l'errore umano e riduce l'impegno degli operatori.
- Apporta modifiche frequenti, piccole e reversibili: progetta carichi di lavoro scalabili e con accoppiamento debole per l'aggiornamento regolare dei componenti. Le tecniche di implementazione automatizzate insieme a modifiche incrementali più piccole riducono il raggio di esplosione, ovvero l'entità dell'impatto, e consentono un'inversione più rapida in caso di guasti. Ciò aumenta la fiducia necessaria per apportare modifiche strategiche al carico di lavoro mantenendo la qualità e adattandosi rapidamente ai cambiamenti delle condizioni di mercato.
- Perfeziona frequentemente le procedure operative: man mano che potenzi i carichi di lavoro, perfeziona le operazioni in modo appropriato. Se usi procedure operative, cerca delle opportunità per migliorarle. Organizza regolari revisioni per accertarti che tutte le procedure siano efficaci e che

i team le conoscano adeguatamente. Se vengono individuate delle lacune, aggiorna le procedure di conseguenza. Comunica gli aggiornamenti procedurali a tutte le parti interessate e ai team. Converti le operazioni in gioco per condividere le best practice e fornire occasioni di formazione ai team.

- Prevedi gli insuccessi: massimizza il successo operativo creando scenari di errore per comprendere il profilo di rischio del carico di lavoro e l'impatto sui risultati aziendali. Testa l'efficacia delle procedure e la risposta del team a questi errori simulati. Prendi decisioni informate per gestire i rischi aperti identificati tramite i test.
- Impara da metriche ed eventi operativi: favorisci il miglioramento tramite le lezioni apprese da tutti gli eventi e gli errori operativi. Condividi ciò che hai imparato con i vari team e con tutta l'organizzazione. Gli insegnamenti evidenziano dati e aneddoti su come le operazioni contribuiscono al conseguimento dei risultati aziendali.
- Usa servizi gestiti: riduci il carico operativo utilizzando servizi gestiti da AWS, laddove possibile. Sviluppa procedure operative basate sulle interazioni con tali servizi.

Definizione

Esistono quattro aree di best practice per l'eccellenza operativa nel cloud:

- Organizzazione
- Preparazione
- Opera
- Evoluzione

La leadership dell'organizzazione definisce gli obiettivi aziendali. La tua organizzazione deve comprendere i requisiti e le priorità e utilizzarli per organizzare e condurre attività a supporto del raggiungimento dei risultati aziendali. Il carico di lavoro deve generare le informazioni necessarie per supportarlo. L'implementazione di servizi per ottenere l'integrazione, il deployment e la distribuzione del carico di lavoro, darà vita a un flusso maggiore di modifiche vantaggiose in fase di produzione attraverso l'automazione dei processi ripetitivi.

Potrebbero esserci rischi inerenti al funzionamento del carico di lavoro. Occorre comprendere questi rischi e prendere una decisione consapevole prima di passare alla fase di produzione. I team devono essere in grado di supportare il carico di lavoro. Le metriche aziendali e operative derivate dai risultati aziendali desiderati ti aiuteranno a comprendere lo stato del carico di lavoro e le attività operative

e di rispondere agli incidenti. Le priorità cambieranno di pari passo con l'evoluzione delle esigenze aziendali e dell'ambiente aziendale. Utilizza questi aspetti come ciclo di feedback per apportare continui miglioramenti all'organizzazione e alle operazioni legate al carico di lavoro.

Best practice

> **Note**
>
> Tutte le domande sull'eccellenza operativa hanno il prefisso OPS come abbreviazione del principio.

Argomenti

Organizzazione

È necessario che i team abbiano una comprensione condivisa dell'intero carico di lavoro, del loro ruolo rispetto al carico di lavoro, nonché degli obiettivi aziendali condivisi. In questo modo potranno stabilire le priorità che possono favorire il successo aziendale. Un'adeguata definizione delle priorità massimizzerà i risultati dei tuoi sforzi. Valuta le esigenze dei clienti interni ed esterni coinvolgendo i principali stakeholder, compresi i team aziendali, di sviluppo e operativi, per stabilire dove concentrare le attività operative. Valutando le esigenze dei clienti otterrai una conoscenza approfondita del supporto necessario per raggiungere i risultati aziendali. Accertati di essere a conoscenza delle linee guida o degli obblighi definiti dalla governance organizzativa e da fattori esterni, come i requisiti di conformità normativa e gli standard di settore, che possono imporre o accentuare un'attenzione specifica. Accertati di disporre di meccanismi per identificare le modifiche ai requisiti di governance interna e di conformità esterni. Se non viene identificato alcun requisito, conferma l'applicazione della due diligence per giungere a tale determinazione. Rivedi regolarmente le tue priorità in modo che possano essere aggiornate al mutare delle esigenze.

Valuta le minacce per il business (ad esempio rischi e responsabilità aziendali e minacce alla sicurezza delle informazioni) e conserva queste informazioni in un registro dei rischi. Valuta l'impatto dei rischi e dei compromessi tra interessi concorrenti o approcci alternativi. Ad esempio, accelerare

l'introduzione sul mercato di nuove funzionalità può essere preferibile all'ottimizzazione dei costi. Oppure, è possibile scegliere un database relazionale per i dati non relazionali per semplificare l'iniziativa di migrazione di un sistema senza refactoring. Gestisci i vantaggi e i rischi per prendere decisioni informate nel determinare dove concentrare gli sforzi. Alcuni rischi o scelte possono essere accettabili per un certo periodo di tempo, potrebbe essere possibile ridurre i rischi associati o la presenza di un rischio potrebbe diventare inaccettabile, nel qual caso si intraprenderà un'azione per risolverlo.

I tuoi team devono comprendere quale contributo offrono nel raggiungimento dei risultati aziendali. I team devono avere obiettivi condivisi e comprendere il proprio ruolo nel successo degli altri team. Comprendere la responsabilità, la proprietà, il modo in cui vengono prese le decisioni e chi ha l'autorità decisionale aiuterà a concentrare gli sforzi e a ottimizzare i contributi dei team. Le esigenze di un team sono influenzate dal cliente supportato, dall'organizzazione, dalla composizione del team e dalle caratteristiche del carico di lavoro. Non è ragionevole aspettarsi che un singolo modello operativo sia in grado di supportare tutti i team e i relativi carichi di lavoro dell'organizzazione.

Assicurati che siano identificati i responsabili di ogni applicazione, carico di lavoro, piattaforma e componente dell'infrastruttura e che per ogni processo e procedura sia identificato un responsabile della definizione e dei responsabili delle prestazioni.

La comprensione del valore aziendale di ogni componente, processo e procedura, del motivo per cui tali risorse sono presenti o le attività vengono eseguite e del perché tale proprietà esiste indirizzerà le azioni dei membri del team. Definisci chiaramente le responsabilità dei membri del team in modo che possano agire in modo appropriato e disporre di meccanismi per identificare responsabilità e proprietà. Implementa meccanismi per richiedere aggiunte, modifiche ed eccezioni in modo da non porre limiti all'innovazione. Definisci gli accordi tra i team che descrivono il modo in cui collaborano per supportarsi reciprocamente e contribuire ai risultati aziendali.

Fornisci supporto ai membri del team in modo che possano essere più efficaci nell'azione e nel supporto dei risultati aziendali. La leadership aziendale di alto livello deve stabilire le aspettative e misurare il successo. Gli alti dirigenti sono promotori, sostenitori e motori per l'adozione delle best practice e l'evoluzione dell'organizzazione. Consenti ai membri del team di intervenire quando i risultati sono a rischio per ridurre al minimo l'impatto e incoraggiali a rivolgersi ai responsabili decisionali e alle parti interessate quando ritengono che esista un rischio, in modo da poterlo risolvere e prevenire gli incidenti. Fornisci comunicazioni tempestive, chiare e concrete dei rischi noti e degli eventi pianificati in modo che i membri del team possano agire in modo tempestivo e appropriato.

Incoraggia la sperimentazione per accelerare l'apprendimento e mantenere i membri del team interessati e coinvolti. I team devono aumentare le proprie competenze per adottare nuove

tecnologie e supportare i cambiamenti della domanda e delle responsabilità. Fornisci il tuo supporto e incoraggiamento offrendo tempo strutturato dedicato per l'apprendimento. Assicurati che i membri del team dispongano delle risorse, in termini sia di strumenti sia di membri del team, per avere successo e adattarsi, sostenendo i risultati aziendali. Sfrutta la diversità tra organizzazioni per cercare più prospettive uniche. Usa questa prospettiva per incrementare l'innovazione, mettere in discussione le tue ipotesi e ridurre il rischio di conferme parziali. Aumenta l'inclusione, la diversità e l'accessibilità all'interno dei team per ottenere prospettive vantaggiose.

Se esistono requisiti normativi e di conformità esterni applicabili alla tua organizzazione, utilizza le risorse fornite da AWS Cloud Compliance per promuovere la formazione dei tuoi team affinché siano in grado di valutare il relativo impatto sulle tue priorità. Il Canone di architettura enfatizza l'apprendimento, la misurazione e il miglioramento. Fornisce una strategia coerente per la valutazione delle architetture e l'implementazione di progetti in grado di dimensionarsi nel corso del tempo. AWS mette a disposizione AWS Well-Architected Tool per aiutarti ad analizzare il tuo approccio prima dello sviluppo e lo stato dei tuoi carichi di lavoro prima e durante la fase di produzione. Puoi confrontare i carichi di lavoro con le best practice architetturali AWS più recenti, monitorarne lo stato generale e ottenere informazioni sui potenziali rischi. AWS Trusted Advisor è uno strumento che fornisce l'accesso a una serie di controlli di base che propongono ottimizzazioni utili per la definizione delle tue priorità. I clienti del supporto Business ed Enterprise hanno accesso a ulteriori controlli a livello di sicurezza, affidabilità, prestazioni, ottimizzazione dei costi e sostenibilità che possono essere utili per definire le loro priorità.

AWS può aiutarti a istruire i tuoi team su AWS e i suoi servizi, affinché comprendano meglio in che modo le loro scelte possono influire sul carico di lavoro. Per istruire i tuoi team, è consigliabile utilizzare le risorse fornite da AWS Support (AWS Knowledge Center, AWS Discussion Forum e AWS Support Center) e la documentazione AWS. Raggiungi AWS Support attraverso il AWS Support Center per assistenza sulle tue domande su AWS. AWS condivide inoltre le best practice e i modelli appresi attraverso la gestione di AWS nella Amazon Builders' Library. Un'ampia gamma di altre informazioni utili è disponibile tramite il blog AWS e il podcast ufficiale di AWS. AWS Training and Certification offre risorse di formazione tramite corsi digitali gestiti dall'utente sulle nozioni di base di AWS. Per supportare ulteriormente lo sviluppo delle competenze AWS del tuo team, è anche possibile iscriversi a corsi di formazione con istruttore.

Per facilitare la gestione dei modelli operativi, è consigliabile utilizzare strumenti o servizi che consentano di gestire centralmente gli ambienti su più account, ad esempio AWS Organizations. Servizi come AWS Control Tower ampliano questa funzionalità di gestione consentendoti di definire piani (a supporto dei tuoi modelli operativi) per configurare gli account, applicare la governance continua tramite AWS Organizations e automatizzare il provisioning di nuovi account. I fornitori di

servizi gestiti, come AWS Managed Services, AWS Managed Services Partners o i fornitori di servizi gestiti della AWS Partner Network offrono esperienza nell'implementazione di ambienti cloud e supportano i requisiti di sicurezza e conformità e gli obiettivi aziendali. L'aggiunta di servizi gestiti al tuo modello operativo ti consente di risparmiare tempo e risorse e ti permette di mantenere i team interni snelli e focalizzati sui risultati strategici che differenzieranno la tua attività, anziché sullo sviluppo di nuove competenze e funzionalità.

Le seguenti domande si concentrano su queste considerazioni relative all'eccellenza operativa. (Per l'elenco completo delle domande e delle best practice relative all'eccellenza operativa, consulta l' Appendice).

OPS 1 In che modo stabilisci quali sono le tue priorità?
È necessario che ognuno comprenda il proprio ruolo nel conseguimento del successo aziendale . Devi disporre di obiettivi comuni al fine di stabilire le priorità per le risorse. Ciò massimizzerà i risultati dei tuoi sforzi.

OPS 2 In che modo strutturi la tua organizzazione per supportare i risultati aziendali?
I tuoi team devono comprendere quale contributo offrono nel raggiungimento dei risultati aziendali. I team devono avere obiettivi condivisi e comprendere il proprio ruolo nel successo degli altri team. Comprendere la responsabilità, la proprietà, il modo in cui vengono prese le decisioni e chi ha l'autorità decisionale aiuterà a concentrare gli sforzi e a ottimizzare i contributi dei team.

OPS 3 In che modo la cultura aziendale supporta i risultati aziendali?
Fornisci supporto ai membri del team in modo che possano essere più efficaci nell'azione e nel supporto dei risultati aziendali.

Ad esempio, a un certo punto potresti realizzare che desideri dare maggiore risalto a un piccolo sottoinsieme delle tue priorità. Utilizza un approccio equilibrato nel lungo termine per garantire lo sviluppo delle capacità necessarie e la gestione del rischio. Rivedi regolarmente le tue priorità e aggiornale al mutare delle esigenze. Quando la responsabilità e la proprietà sono indefinite o sconosciute, rischi sia di non affrontare tempestivamente le attività necessarie sia di adoperarti in

modo ridondante e potenzialmente conflittuale per rispondere a tali esigenze. La cultura organizzativa influisce direttamente sulla soddisfazione sul lavoro e sulla conservazione dei membri del team. Sostieni il coinvolgimento e le capacità dei membri del tuo team per ottenere il successo della tua attività. La sperimentazione è necessaria per realizzare l'innovazione e trasformare le idee in risultati. Un risultato indesiderato è un esperimento riuscito che ha identificato un percorso che non porterà al successo.

Preparazione

Per prepararti all'eccellenza operativa devi comprendere i carichi di lavoro e i loro comportamenti previsti. Sarai dunque in grado di progettare i carichi di lavoro in modo tale che forniscano informazioni sul loro stato e di creare le procedure per supportarli adeguatamente.

Progetta il tuo carico di lavoro affinché ti fornisca le informazioni necessarie a comprenderne lo stato interno (ad esempio, parametri, log, eventi e tracce) in tutti i componenti a supporto dell'osservabilità e dell'analisi dei problemi. L'osservabilità va oltre il semplice monitoraggio, in quanto fornisce una comprensione completa del funzionamento interno di un sistema basata sui suoi output esterni. L'osservabilità è legata a doppio filo a metriche, log e tracce per offrire informazioni approfondite sul comportamento e sulle dinamiche del sistema. Grazie a un'osservabilità efficace, i team possono distinguere modelli, anomalie e tendenze, così da essere in grado di affrontare in modo proattivo potenziali problemi e mantenere l'integrità del sistema. L'identificazione degli indicatori chiave di prestazione (KPI) è fondamentale per garantire l'allineamento tra le attività di monitoraggio e gli obiettivi aziendali. Questo allineamento garantisce che i team prendano decisioni basate sui dati e su metriche realmente importanti, ottimizzando sia le prestazioni del sistema sia i risultati aziendali. Inoltre, l'osservabilità consente alle aziende di essere proattive anziché reattive. I team possono comprendere le relazioni causa-effetto all'interno dei loro sistemi, prevedendo e prevenendo i problemi anziché limitarsi a reagire quando si verificano. Con l'evolversi dei carichi di lavoro, è essenziale riesaminare e perfezionare la strategia di osservabilità, assicurandosi che rimanga pertinente ed efficace.

Adotta strategie che migliorino il flusso delle modifiche in produzione e che consentano la rifattorizzazione, il feedback veloce sulla qualità e la correzione di errori. Tali prassi accelerano l'ingresso in produzione delle modifiche vantaggiose, limitano i problemi distribuiti e consentono una rapida identificazione e risoluzione dei problemi introdotti attraverso le attività di distribuzione o scoperti negli ambienti.

Adotta prassi per fornire un feedback rapido sulla qualità e che permettano un ripristino veloce dalle modifiche che non hanno i risultati previsti. L'uso di queste prassi consente di mitigare l'impatto

dei problemi introdotti attraverso la distribuzione delle modifiche. Prepara un piano in caso di esito negativo delle modifiche in modo da poter rispondere più rapidamente se necessario, testando e convalidando le modifiche apportate. Sii consapevole delle attività pianificate nei tuoi ambienti in modo da poter gestire il rischio di modifiche che influiscono sulle attività pianificate. Privilegia le modifiche frequenti, piccole e reversibili per limitarne l'ambito. Semplificherai così la risoluzione dei problemi, accelerando la correzione e mantenendo la possibilità di rollback delle modifiche. In tal modo, è anche possibile ottenere più frequentemente i vantaggi offerti dalle modifiche importanti.

Valuta la prontezza operativa del carico di lavoro, dei processi e delle procedure, nonché del personale, per comprendere i rischi operativi correlati al carico di lavoro. È consigliabile utilizzare un processo omogeneo (inclusi elenchi di controllo manuali o automatici) per sapere quando puoi rilasciare un carico di lavoro o una modifica. Questo inoltre ti aiuterà a trovare le eventuali aree che necessitano di pianificazioni. Predisponi istruzioni che documentano le tue attività di routine e manuali che guidano i processi per la risoluzione dei problemi. Analizza i vantaggi e i rischi per prendere decisioni informate e consentire l'adozione delle modifiche nella produzione.

In AWS, puoi vedere il tuo carico di lavoro completo (applicazioni, infrastruttura, policy, governance e operazioni) in forma di codice. In tal modo è possibile applicare la stessa disciplina ingegneristica utilizzata per il codice dell'applicazione a ogni elemento dello stack, condividendoli tra team o organizzazioni per sfruttare al massimo i vantaggi delle attività di sviluppo. Utilizza le operazioni come codice nel cloud e sfrutta la possibilità di sperimentare per sviluppare il tuo carico di lavoro e le procedure operative ed esercitarti con gli errori in modo sicuro. AWS CloudFormation ti consente di avere ambienti di sviluppo, di prova e di produzione sandbox, omogenei e basati su modelli, con livelli crescenti di controllo operativo.

Le seguenti domande si concentrano su queste considerazioni relative all'eccellenza operativa.

OPS 4 In che modo implementi l'osservabilità nel carico di lavoro?

Implementare l'osservabilità nel carico di lavoro ti permette di comprendere lo stato di quest'ultimo e di adottare decisioni basate sui dati e che riflettono i requisiti aziendali.

OPS 5 In che modo riduci i difetti, favorisci la correzione e migliori il flusso nella produzione?

Adotta strategie che migliorino il flusso delle modifiche in produzione e che favoriscano la rifattori zzazione, il feedback veloce sulla qualità e la correzione di errori. Tali prassi accelerano l'ingresso in produzione delle modifiche vantaggiose, contengono i problemi che si sono diffusi e permetton

OPS 5 In che modo riduci i difetti, favorisci la correzione e migliori il flusso nella produzione?

o di ottenere una rapida identificazione e risoluzione dei problemi introdotti attraverso le attività di implementazione.

OPS 6 In che modo mitighi i rischi della distribuzione?

Adotta prassi per fornire un feedback rapido sulla qualità e che permettano un ripristino veloce dalle modifiche che non hanno i risultati previsti. L'uso di queste prassi consente di mitigare l'impatto dei problemi introdotti attraverso la distribuzione delle modifiche.

OPS 7 Come fai a sapere che sei pronto a supportare un carico di lavoro?

Valuta la disponibilità operativa del carico di lavoro, dei processi e delle procedure, nonché del personale per comprendere i rischi operativi correlati al carico di lavoro.

Investi nell'implementazione di attività operative come codice per aumentare al massimo la produttività del personale operativo, ridurre al minimo la frequenza degli errori e consentire risposte automatizzate. Utilizza l'analisi prefallimentare per prevedere errori e creare procedure ove opportuno. Applica i metadati utilizzando i tag delle risorse e i AWS Resource Groups seguendo una strategia di applicazione dei tag coerente per consentire l'identificazione delle risorse. Applica tag alle risorse per organizzare, monitorare i costi e controllare gli accessi e ottimizza l'esecuzione delle attività operative automatizzate. Adotta procedure di distribuzione che sfruttino l'elasticità del cloud per facilitare le attività di sviluppo e la pre-distribuzione dei sistemi e avere implementazioni più rapide. Quando apporti modifiche agli elenchi di controllo che utilizzi per valutare i tuoi carichi di lavoro, pianifica quello che farai con i sistemi live che non risultano più conformi.

Operatività

L'osservabilità ti consente di concentrarti su dati significativi e di comprendere le interazioni e l'output del tuo carico di lavoro. Concentrandoti sugli approfondimenti essenziali ed eliminando i dati non necessari, mantieni un approccio diretto alla comprensione delle prestazioni del carico di lavoro. È essenziale non solo raccogliere dati, ma anche interpretarli correttamente. Definisci linee guida chiare, imposta soglie di avviso appropriate e monitora attivamente eventuali deviazioni. Un cambiamento in una metrica chiave, specialmente se correlata ad altri dati, permette di individuare

aree problematiche specifiche. Grazie all'osservabilità hai strumenti per prevedere e affrontare potenziali sfide, assicurando che il tuo carico di lavoro funzioni senza intoppi e soddisfi le esigenze aziendali.

La corretta operatività di un carico di lavoro è misurata dal raggiungimento di risultati per l'azienda e per i clienti. Definisci i risultati desiderati, determina in che modo verrà misurato il successo e individua i parametri che saranno usati nei calcoli per determinare se il carico di lavoro e le operazioni sono efficaci. L'integrità delle operazioni include sia lo stato del carico di lavoro sia lo stato e il successo delle operazioni a supporto del carico di lavoro (ad esempio, la distribuzione e la risposta agli incidenti). Stabilisci le basi dei parametri per migliorare, eseguire indagini e intervenire, raccogliere e analizzare i parametri, quindi conferma la tua comprensione del successo operativo e della sua evoluzione nel corso del tempo. Usa i parametri raccolti per determinare il grado di soddisfazione dei clienti, capire se stai rispondendo alle esigenze aziendali e individuare gli aspetti da migliorare.

La gestione efficiente ed efficace degli eventi operativi è fondamentale per raggiungere l'eccellenza operativa. Ciò si applica agli eventi operativi sia pianificati che non. Usa istruzioni precise per gli eventi chiari e ricorri ai manuali per favorire l'analisi e la risoluzione degli altri eventi. Attribuisci la priorità alle risposte agli eventi in base al loro impatto sull'azienda e sui clienti. Assicurati che, in caso di avvisi in risposta a un evento, vi sia una procedura associata da seguire, con un proprietario ben preciso. Definisci in anticipo il personale richiesto per risolvere un evento e includi dei processi di escalation per coinvolgere altro personale, ove necessario, in base all'urgenza e all'impatto. Individua e coinvolgi le persone che hanno l'autorità per prendere decisioni in merito alle linee d'azione laddove vi sia un impatto aziendale dovuto a una risposta a un evento non gestito precedentemente.

Comunica lo stato operativo dei carichi di lavoro tramite pannelli di controllo e notifiche personalizzati in base al pubblico di destinazione (ad esempio cliente, azienda, sviluppatori, addetti alle operazioni), in modo che gli interessati possano agire in maniera adeguata, che le loro aspettative vengano soddisfatte e che siano informati sulla ripresa delle normali operazioni.

In AWS puoi generare panoramiche di pannelli di controllo per i parametri raccolti dai carichi di lavoro e in modo nativo da AWS. Puoi sfruttare CloudWatch o applicazioni di terze parti per aggregare e presentare panoramiche a livello di business, di carico di lavoro e di operazioni delle attività operative. AWS fornisce approfondimenti sui carichi di lavoro attraverso funzionalità di logging, tra cui AWS X-Ray, CloudWatch, CloudTrail e VPC Flow Logs, che consentono di identificare i problemi del carico di lavoro a supporto dell'analisi delle cause principali e della risoluzione dei problemi.

Le seguenti domande si concentrano su queste considerazioni relative all'eccellenza operativa.

OPS 8 In che modo utilizzi l'osservabilità del carico di lavoro nella tua organizzazione?

Garantire l'integrità del carico di lavoro sfruttando l'osservabilità. Utilizzare metriche, log e tracce pertinenti per ottenere una visione completa delle prestazioni del carico di lavoro e risolvere i problemi in modo efficiente.

OPS 9 Come fai a comprendere lo stato delle operazioni?

Definisci, acquisisci e analizza i parametri delle operazioni per ottenere visibilità sugli eventi delle operazioni, in modo da intraprendere le azioni appropriate.

OPS 10 In che modo gestisci gli eventi del carico di lavoro e delle operazioni?

Prepara e convalida le procedure in risposta agli eventi per ridurre al minimo il loro impatto sul tuo carico di lavoro.

Tutti i parametri raccolti devono essere allineati alle esigenze aziendali e ai risultati che supportano. Sviluppa risposte con script per eventi ben compresi e automatizza le prestazioni in risposta al riconoscimento dell'evento.

Evoluzione

Impara, condividi e migliora continuamente per sostenere l'eccellenza operativa. Dedica dei cicli di lavoro al raggiungimento di miglioramenti incrementali quasi continui. Esegui l'analisi post-incidente di tutti gli eventi che influiscono sul cliente. Identifica i fattori che contribuiscono e le azioni preventive per limitare o prevenire la ricorrenza. Comunica i fattori che contribuiscono alle comunità interessate, nel modo più adeguato. Valuta regolarmente e assegna le priorità alle opportunità di miglioramento (ad esempio, richieste di funzionalità, risoluzione dei problemi e requisiti di conformità), includendo sia il carico di lavoro sia le procedure operative.

Includi i loop di feedback nelle tue procedure per individuare rapidamente gli aspetti che devono essere migliorati e per acquisire conoscenze dall'esecuzione delle operazioni.

Condividi le lezioni apprese con i vari team per condividerne anche i vantaggi. Analizza le tendenze all'interno delle lezioni apprese ed esegui analisi trasversali retrospettive dei parametri operativi per

individuare le opportunità e i metodi di miglioramento. Implementa le modifiche previste per garantire il miglioramento e valuta i risultati per favorire il successo.

In AWS, è possibile esportare i dati di log in Amazon S3 o inviare log direttamente ad Amazon S3 per lo storage a lungo termine. Utilizzando AWS Glue, è possibile individuare e preparare i dati di log in Amazon S3 per l'analisi, archiviando i metadati associati in AWS Glue Data Catalog. Amazon Athena, grazie all'integrazione nativa con AWS Glue, può essere quindi utilizzato per analizzare i dati di log, eseguendo query tramite SQL standard. Utilizzando uno strumento di business intelligence come Amazon QuickSight puoi visualizzare, esplorare e analizzare i tuoi dati. Rilevamento di tendenze ed eventi di interesse che possono portare a miglioramenti.

La seguente domanda si concentra su queste considerazioni relative all'eccellenza operativa.

OPS 11 In che modo fai evolvere le operazioni?

Dedica tempo e risorse per ottenere un miglioramento incrementale pressoché continuo, per far evolvere l'efficacia e l'efficienza delle tue operazioni.

L'evoluzione efficace delle operazioni si basa sugli elementi seguenti: miglioramenti piccoli ma frequenti; creazione di ambienti sicuri e tempo per sperimentare, sviluppare e testare i miglioramenti; ambienti in cui le persone siano incoraggiate a imparare dagli errori. Il supporto alle operazioni per ambienti sandbox, di sviluppo, di prova e di produzione, con un crescente livello di controlli operativi, facilita lo sviluppo e aumenta la prevedibilità dei risultati positivi dalle modifiche passate in produzione.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle best practice relative all'eccellenza operativa.

Documentazione

- DevOps e AWS

Whitepaper

- Principio dell'eccellenza operativa

Video

- DevOps di Amazon

Sicurezza

Il pilastro della sicurezza contempla la capacità di proteggere dati, sistemi e asset per sfruttare le tecnologie cloud in modo da migliorare la sicurezza.

Il principio della sicurezza offre una panoramica dei principi di progettazione, delle best practice e delle domande. È possibile trovare linee guida prescrittive sull'implementazione nel Whitepaper sul principio della sicurezza.

Argomenti

- Principi di progettazione
- Definizione
- Best practice
- Risorse

Principi di progettazione

Esistono sette principi di progettazione per la sicurezza nel cloud:

- Implementa una solida base identitariaimplementa il principio del privilegio minimo e attua la separazione dei compiti con la corretta autorizzazione per ciascuna interazione con le risorse AWS. Centralizza la gestione delle identità e mira a eliminare la dipendenza dalle credenziali statiche a lungo termine.
- Abilita la tracciabilità: monitora, avvisa e verifica le azioni e le modifiche al tuo ambiente in tempo reale. Integra la raccolta di log e parametri con i sistemi per analizzare e intervenire automaticamente.
- Applica la sicurezza a tutti i livelli: applica un approccio di difesa avanzata con più controlli di sicurezza. Applicalo a tutti i livelli (ad esempio, edge di rete, VPC, bilanciamento del carico, ogni istanza e servizio di elaborazione, sistema operativo, applicazione e codice).
- Automatizza le best practice per la sicurezza: meccanismi di sicurezza automatici basati sul software migliorano la capacità di ricalibrare il sistema in modo sicuro, più rapido e conveniente.

Crea architetture sicure, compresa l'implementazione dei controlli, che sono definite e gestite come codice nei modelli controllati dalle versioni.

- Proteggi i dati in transito e a riposo: classifica i dati secondo livelli di sensibilità e meccanismi d'uso, come crittografia, tokenizzazione e controllo di accesso, ove opportuno.
- Tieni le persone a distanza dai dati: utilizza meccanismi e strumenti per ridurre o eliminare l'esigenza di accesso diretto o di elaborazione manuale dei dati. Ciò riduce il rischio di perdita, modifica e altri errori umani durante la gestione dei dati sensibili.
- Preparati per gli eventi di sicurezza: preparati per un incidente ipotetico creando policy e processi di gestione degli incidenti allineati ai requisiti dell'organizzazione. Esegui simulazioni di risposta agli incidenti e utilizza strumenti dotati di automazione per aumentare la velocità nel rilevamento, nell'indagine e nel ripristino.

Definizione

Esistono sei aree di best practice per la sicurezza nel cloud:

- Sicurezza
- Gestione di identità e accessi (Identity and access management)
- Rilevamento
- Protezione dell'infrastruttura
- Protezione dei dati
- Risposta agli incidenti

Prima di progettare qualsiasi carico di lavoro, è necessario implementare pratiche che influenzano la sicurezza. Dovrai controllare chi può fare cosa. Inoltre, devi essere in grado di identificare gli incidenti di sicurezza, proteggere i tuoi sistemi e i tuoi servizi e mantenere la riservatezza e l'integrità dei dati attraverso la loro protezione. Dovresti avere dei processi ben definiti e rodati per rispondere a eventuali problemi di sicurezza. Questi strumenti e tecniche sono importanti perché supportano obiettivi come la prevenzione delle perdite finanziarie o la conformità agli obblighi normativi.

Il modello di responsabilità condivisa di AWS permette alle organizzazioni che adottano il cloud di raggiungere i loro obiettivi in termini di sicurezza e conformità. Dato che AWS mette fisicamente in sicurezza l'infrastruttura che supporta i nostri servizi cloud, come cliente AWS puoi concentrarti sull'utilizzo dei servizi per raggiungere gli obiettivi. Il cloud AWS fornisce, inoltre, l'accesso ai dati sulla sicurezza e offre un approccio automatico per rispondere agli eventi di sicurezza.

Best practice

Argomenti

Sicurezza

Per gestire il carico di lavoro in modo sicuro, è necessario applicare le best practice globali a ogni area di sicurezza. Segui i requisiti e i processi definiti in termini di eccellenza operativa a livello organizzativo e di carico di lavoro e applicali a tutte le aree.

Rimanere aggiornati con le raccomandazioni di AWS e del settore nonché con l'intelligence sulle minacce aiuta a sviluppare il modello di rischio e gli obiettivi di controllo. L'automazione dei processi di sicurezza, i test e la convalida consentono di ricalibrare le operazioni di sicurezza.

La seguente domanda si concentra su queste considerazioni relative alla sicurezza (per l'elenco completo delle domande e delle best practice, consulta l' Appendice.).

SEC 1 Come gestisci in modo sicuro un carico di lavoro?

Per gestire il carico di lavoro in modo sicuro, è necessario applicare le best practice globali a ogni area di sicurezza. Segui i requisiti e i processi definiti in termini di eccellenza operativa a livello organizzativo e di carico di lavoro e applicali a tutte le aree. Rimanere aggiornati con le raccomand azioni di AWS, le fonti di settore nonché con l'intelligence sulle minacce aiuta a sviluppare il modello di rischio e gli obiettivi di controllo. L'automazione dei processi di sicurezza, i test e la convalida consentono di ricalibrare le operazioni di sicurezza.

In AWS, è consigliabile separare i diversi carichi di lavoro per account, in base alla loro funzione e ai requisiti di conformità o di sensibilità dei dati.

Gestione di identità e accessi

La gestione delle identità e degli accessi è una parte principale di un programma di sicurezza delle informazioni e garantisce che solo gli utenti e i componenti autorizzati e autenticati possano accedere alle tue risorse e solo nella modalità che hai stabilito. Ad esempio, è necessario definire i principali (ovvero account, utenti, ruoli e servizi che possono eseguire operazioni nel tuo account), creare policy allineate a tali principali e implementare una forte gestione delle credenziali. Questi elementi a gestione privilegiata formano i concetti chiave dell'autenticazione e dell'autorizzazione.

In AWS, la gestione dei privilegi è principalmente supportata dal servizio AWS Identity and Access Management (IAM), che consente di controllare l'accesso utente e l'accesso programmatico ai servizi e alle risorse AWS. È necessario applicare criteri granulari che assegnano autorizzazioni a un utente, gruppo, ruolo o risorsa. Hai anche la possibilità di richiedere pratiche di password complesse, come il livello di complessità, evitare il riutilizzo e applicare l'autenticazione a più fattori (MFA). È possibile utilizzare la federazione con il servizio di directory esistente. Per i carichi di lavoro che richiedono che i sistemi abbiano accesso ad AWS, IAM consente l'accesso sicuro tramite ruoli, profili dell'istanza, federazione delle identità e credenziali temporanee.

Le seguenti domande si concentrano su queste considerazioni relative alla sicurezza.

SEC 2 Come gestisci l'autenticazione per persone e macchine?

Esistono due tipi di identità che è necessario gestire quando si utilizzano carichi di lavoro AWS sicuri. Comprendere il tipo di identità necessaria per gestire e concedere l'accesso ti aiuta a garantire che le identità corrette abbiano accesso alle risorse giuste nelle condizioni adeguate.

Identità umane: amministratori, sviluppatori, operatori e utenti finali necessitano di un'identità per accedere agli ambienti e alle applicazioni AWS. Si tratta di membri dell'organizzazione o utenti esterni con cui collabori e che interagiscono con le tue risorse AWS tramite browser Web, applicazioni client o strumenti a riga di comando interattivi.

Identità di macchine: le applicazioni di servizio, gli strumenti operativi e i carichi di lavoro necessita no di un'identità per effettuare richieste ai servizi AWS, ad esempio per leggere i dati. Queste identità includono macchine in esecuzione nell'ambiente AWS, ad esempio istanze Amazon EC2 o funzioni AWS Lambda. Puoi gestire le identità di macchine anche per soggetti esterni che necessitano dell'accesso. Inoltre, potresti disporre di macchine al di fuori di AWS che devono accedere al tuo ambiente AWS.

SEC 3 Come gestisci le autorizzazioni per persone e macchine?

Gestisci le autorizzazioni per controllare l'accesso alle identità di persone e macchine che richiedono l'accesso ad AWS e al tuo carico di lavoro. Le autorizzazioni controllano chi può accedere a cosa e a quali condizioni.

Le credenziali non devono essere condivise tra nessun utente o sistema. L'accesso degli utenti dovrebbe essere concesso utilizzando un approccio con privilegi minimi con le migliori pratiche, inclusi i requisiti di password e l'applicazione del MFA. L'accesso programmatico, comprese le chiamate API ai servizi AWS, deve essere eseguito utilizzando credenziali temporanee e con privilegi limitati come quelle emesse da AWS Security Token Service.

AWS offre risorse che possono aiutarti nella gestione dell'identità e degli accessi. Per apprendere le best practice, esplora i nostri corsi pratici sulla gestione delle credenziali e dell'autenticazione, sul controllo dell'accesso umanoe sul controllo dell'accesso programmatico.

Rilevamento

Puoi utilizzare i controlli di rilevamento per identificare una potenziale minaccia o un potenziale incidente di sicurezza. Questi controlli sono una parte essenziale dei framework di governance e possono essere utilizzati per supportare il processo di qualità o un obbligo legale o di conformità e per l'identificazione delle minacce e gli sforzi nelle risposte. Ci sono diversi tipi di controlli di rilevamento. Ad esempio, la realizzazione di un inventario di risorse e dei loro attributi dettagliati promuove le decisioni più efficienti (e i controlli del ciclo di vita) per stabilire delle baseline operative. Puoi anche utilizzare audit interni, una verifica dei controlli relativi ai sistemi di informazioni, per assicurarti che le practice rispettino le policy e i requisiti e che tu abbia un set corretto di notifiche di avviso automatiche basate sulle condizioni definite. Questi controlli sono fattori di reazione importanti che possono aiutare la tua organizzazione a identificare e capire la portata dell'attività anomala.

In AWS, puoi implementare controlli investigativi elaborando registri, eventi e monitoraggio che consentono audit, analisi automatizzate e notifiche. I registri CloudTrail, le chiamate API AWS e CloudWatch forniscono il monitoraggio di parametri con notifiche, mentre AWS Config fornisce la cronologia delle configurazioni. Amazon GuardDuty è un servizio di rilevazione delle minacce che monitora costantemente possibili comportamenti dannosi o non autorizzati, così da proteggere i tuoi account e i tuoi carichi di lavoro su AWS. Sono inoltre disponibili log a livello di servizio, ad esempio puoi utilizzare Amazon Simple Storage Service (Amazon S3) per registrare le richieste di accesso.

La seguente domanda si concentra su queste considerazioni relative alla sicurezza

SEC 4 In che modo individui ed esamini gli eventi di sicurezza?

Acquisisci ed analizza gli eventi a partire da log e parametri per acquistare visibilità. Agisci su eventi di sicurezza e potenziali minacce per contribuire a rendere sicuro il carico di lavoro.

La gestione dei log è una parte importante di un carico di lavoro Well-Architected per ragioni che vanno da requisiti di sicurezza o forensi a disposizioni normative o legali. È fondamentale analizzare i registri e rispondere in modo da identificare potenziali incidenti di sicurezza. AWS offre funzionalità che semplificano l'implementazione della gestione dei registri, offrendo la possibilità di definire un ciclo di vita di conservazione dei dati o di definire dove verranno conservati, archiviati o eventualmente eliminati. Ciò rende la gestione dei dati prevedibile e affidabile, più semplice ed economica.

Protezione dell’infrastruttura

La protezione dell'infrastruttura comprende delle metodologie di controllo, come la difesa approfondita, necessarie per rispettare le best practice e gli obblighi organizzativi e normativi. L'utilizzo di queste metodologie è fondamentale per ottenere operazioni continuative e di successo sia nel cloud che in locale.

In AWS, è possibile implementare l'ispezione di pacchetti con stato e senza stato, sia utilizzando tecnologie native di AWS, sia utilizzando prodotti e servizi dei partner disponibili attraverso Marketplace AWS. È necessario utilizzare Amazon Virtual Private Cloud (Amazon VPC) per creare un ambiente privato, protetto e scalabile in cui è possibile definire la propria topologia, inclusi gateway, tabelle di indirizzamento e sottoreti pubbliche e private.

Le seguenti domande si concentrano su queste considerazioni relative alla sicurezza.

SEC 5 In che modo proteggi le risorse di rete?

Qualsiasi carico di lavoro che abbia una qualche forma di connettività di rete, che si tratti di Internet o di una rete privata, richiede più livelli di difesa per proteggere da minacce esterne e interne basate sulla rete.

SEC 6 In che modo proteggi le risorse di calcolo?

Le risorse di calcolo nel carico di lavoro richiedono più livelli di difesa per contribuire alla protezion e da minacce esterne ed interne. Le risorse di calcolo includono istanze EC2, container, funzioni di AWS Lambda, servizi di database, dispositivi IoT e altro.

Si consigliano più livelli di difesa in qualsiasi tipo di ambiente. Nel caso della protezione dell'infrastruttura, molti concetti e metodi sono validi sia per modelli cloud che in locale. L'applicazione della protezione dei confini, il monitoraggio dei punti di ingresso e di uscita e la registrazione, il monitoraggio e le notifiche completi sono tutti elementi essenziali per un efficace piano di sicurezza delle informazioni.

I clienti AWS sono in grado di adattare o rafforzare la configurazione di Amazon Elastic Compute Cloud (Amazon EC2), di un container Amazon Elastic Container Service (Amazon ECS) o di un'istanza AWS Elastic Beanstalk e mantenere questa configurazione su una Amazon Machine Image (AMI) immutabile. Quindi, che siano attivati da Auto Scaling o lanciati manualmente, tutti i nuovi server virtuali (istanze) lanciati con questa AMI utilizzeranno la configurazione avanzata.

Protezione dei dati

Prima di progettare qualsiasi sistema, devono essere stabiliti i requisiti fondamentali che influenzano la sicurezza. Ad esempio, la classificazione dei dati fornisce un modo per categorizzare i dati organizzativi basati sui livelli di sensibilità, mentre la crittografia protegge i dati evitandone l'intelligibilità per gli accessi non autorizzati. Questi strumenti e tecniche sono importanti perché supportano obiettivi come la prevenzione delle perdite finanziarie o la conformità agli obblighi normativi.

In AWS, le seguenti pratiche facilitano la protezione dei dati:

- Come cliente AWS mantieni il pieno controllo sui tuoi dati.
- AWS semplifica la crittografia dei dati e la gestione delle chiavi, inclusa la rotazione regolare delle chiavi, che può essere facilmente automatizzata da AWS o gestita da te.
- È disponibile la registrazione dettagliata che contiene contenuti importanti, come l'accesso ai file e le modifiche.
- AWS ha progettato sistemi di storage con una resilienza eccezionale. Ad esempio, Amazon S3 Standard, S3 Standard-IA, One Zone-IA S3 e Amazon Glacier sono tutti progettati per offrire una

resistenza degli oggetti del 99,999999999% in un determinato anno. Questo livello di durabilità corrisponde a una perdita media annua prevista dello 0,000000001% di oggetti.

- Il controllo delle versioni, che può far parte di un più ampio processo di gestione del ciclo di vita dei dati, può proteggere da sovrascritture accidentali, eliminazioni e danni simili.
- AWS non avvia mai il trasferimento di dati tra Regioni. Il contenuto inserito in una regione rimarrà in quella regione a meno che tu non abiliti esplicitamente una funzione o utilizzi un servizio che fornisce tale funzionalità.

Le seguenti domande si concentrano su queste considerazioni relative alla sicurezza.

SEC 7 In che modo classifichi i dati?

La classificazione fornisce un modo per categorizzare i dati in base ai livelli di criticità e sensibilità, in modo da aiutarti a determinare i controlli di protezione e conservazione appropriati.

SEC 8 In che modo proteggi i dati inattivi?

Proteggi i dati inattivi implementando più controlli, per ridurre il rischio di accessi non autorizzati o altri comportamenti impropri.

SEC 9 In che modo proteggi i dati in transito?

Proteggi i dati in transito implementando più controlli, per ridurre il rischio di accessi non autorizzati o perdita.

AWS offre molteplici mezzi per crittografare i dati a riposo e in transito. Nei nostri servizi integriamo funzionalità che semplificano la crittografia dei dati. Ad esempio, abbiamo implementato la crittografia lato server (SSE) per Amazon S3 per semplificare l'archiviazione dei dati in forma crittografata. È inoltre possibile disporre che l'intero processo di crittografia e decrittografia HTTPS (generalmente noto come terminazione SSL) sia gestito da Elastic Load Balancing (ELB).

Risposta agli imprevisti

Anche con controlli preventivi e investigativi estremamente maturi, la tua organizzazione dovrebbe comunque attuare processi per rispondere e mitigare il potenziale impatto di incidenti di sicurezza. L'architettura del carico di lavoro influisce fortemente sulla capacità dei team di operare efficacemente durante un incidente, isolare o contenere sistemi e ripristinare le operazioni a uno stato ottimale noto. La messa in atto degli strumenti e l'accesso prima di un incidente di sicurezza e la pratica sistematica della risposta agli incidenti durante i giorni di attività ti aiuterà a garantire che la tua architettura sia in grado di supportare indagini e ripristini tempestivi.

In AWS, le seguenti pratiche facilitano una risposta efficace agli incidenti:

- Sono disponibili registrazioni dettagliate che contengono contenuti importanti, come l'accesso ai file e le modifiche.
- Gli eventi possono essere elaborati automaticamente e possono attivare strumenti che automatizzano le risposte mediante l'uso delle API di AWS.
- Puoi effettuare il pre-provisioning degli strumenti e una "clean room" utilizzando AWS CloudFormation. Questo permette di effettuare indagini forensi in un ambiente sicuro e isolato.

La seguente domanda si concentra su queste considerazioni relative alla sicurezza

SEC 10 In che modo prevedi, reagisci a e risolvi gli incidenti?, rispondi e risolvi gli eventi?

La preparazione è cruciale per un esame tempestivo ed efficace degli incidenti di sicurezza, nonché per la risposta e il ripristino, così da ridurre al minimo potenziali interruzioni dell'orga nizzazione.

Assicurati di poter garantire rapidamente l'accesso al tuo team addetto alla sicurezza e automatizzare l'isolamento delle istanze, oltre che acquisire i dati e lo stato per le indagini forensi.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle best practice relative alla sicurezza.

Documentazione

- Sicurezza del cloud AWS

- Conformità di AWS
- Blog sulla sicurezza di AWS

Whitepaper

- Pilastro della sicurezza
- Panoramica sulla sicurezza di AWS
- Rischio e conformità di AWS

Video

- AWS Security State of the Union
- Panoramica sulla responsabilità condivisa

Affidabilità

Il principio dell'affidabilità comprende la capacità di un carico di lavoro di eseguire la funzione attesa in modo corretto e coerente quando previsto. Include la possibilità di utilizzare e testare il carico di lavoro per tutto il ciclo di vita. Questo documento fornisce linee guida dettagliate sulle best practice per l'implementazione di carichi di lavoro affidabili in AWS.

Il principio dell'affidabilità offre una panoramica dei principi di progettazione, delle best practice e delle domande. È possibile trovare linee guida prescrittive sull'implementazione nel Whitepaper sul principio dell'affidabilità.

Argomenti

- Principi di progettazione
- Definizione
- Best practice
- Risorse

Principi di progettazione

Esistono cinque principi di progettazione per l'affidabilità nel cloud:

- Adotta un approccio di ripristino automatico dagli errori: monitorando gli indicatori chiave di prestazione (KPI) di un carico di lavoro, è possibile attivare l'automazione in caso di superamento di una soglia. Questi KPI dovrebbero essere una misura del valore aziendale, non degli aspetti tecnici del funzionamento del servizio. Ciò consente la notifica e il tracciamento automatici degli errori e i processi di recupero automatizzati che aggirano o riparano l'errore. Con un'automazione più sofisticata è possibile anticipare e correggere gli errori prima che si verifichino.
- Collauda le procedure di ripristino: in un ambiente in locale, spesso vengono eseguiti test per dimostrare che il carico di lavoro funziona in uno scenario specifico. I test non vengono generalmente utilizzati per convalidare le strategie di recupero. Nel cloud, puoi testare il modo in cui il carico di lavoro incorre nell'errore e convalidare le procedure di ripristino. È possibile utilizzare l'automazione per simulare diversi errori o per ricreare scenari che in precedenza hanno portato a errori. Questo approccio presenta percorsi di errore che è possibile testare e correggere prima che si verifichi uno scenario di errore reale, riducendo così il rischio.
- Dimensiona orizzontalmente per aumentare la disponibilità dei carichi di lavoro aggregati: sostituisci una risorsa grande con più risorse piccole per ridurre l'impatto di un singolo guasto sul carico di lavoro complessivo. Distribuisci le richieste su molteplici risorse più piccole per garantire che non condividano un punto di errore comune.
- Smetti di fare congetture sulla capacità: una causa comune di guasti nei carichi di lavoro in locale è la saturazione delle risorse, quando le richieste assegnate a un carico di lavoro superano la capacità di quel carico di lavoro (questo è spesso l'obiettivo di attacchi di tipo Denial of Service). Nel cloud, è possibile monitorare la domanda e l'utilizzo dei carichi di lavoro, nonché automatizzare l'aggiunta o la rimozione di risorse per mantenere il livello ottimale, al fine di soddisfare la domanda senza un provisioning eccessivo o inferiore. Esistono ancora dei limiti, ma alcune quote possono essere controllate e altre possono essere gestite (consulta Gestisci vincoli e Service Quotas).
- Gestisci il cambiamento nell'automazione: le modifiche all'infrastruttura dovrebbero essere apportate utilizzando l'automazione. Le modifiche che devono essere gestite includono le modifiche all'automazione, che possono quindi essere monitorate e revisionate.

Definizione

Esistono quattro aree di best practice per l'affidabilità nel cloud:

- Fondamenti
- Architettura del carico di lavoro
- Gestione delle modifiche

- Gestione degli errori

Per ottenere affidabilità, è necessario iniziare dalle basi: un ambiente in cui le quote di servizio e la topologia di rete sono in grado di supportare il carico di lavoro. L'architettura del carico di lavoro del sistema distribuito deve essere progettata per prevenire e mitigare gli errori. Il carico di lavoro deve gestire le variazioni nella domanda o nei requisiti e deve essere progettato per rilevare l'errore e correggersi automaticamente.

Best practice

Argomenti

Fondamenti

I requisiti di base sono quelli il cui ambito si estende oltre un singolo carico di lavoro o progetto. Prima di progettare qualsiasi sistema, devono essere stabiliti i requisiti fondamentali che influenzano l'affidabilità. Ad esempio, è necessario disporre di una larghezza di banda di rete sufficiente verso il data center.

Con AWS, la maggior parte di questi requisiti di base è già incorporata o può essere affrontata in base alle esigenze. Il cloud è progettato per essere quasi illimitato, perciò è responsabilità di AWS soddisfare i requisiti di capacità di rete e di elaborazione sufficienti, lasciandoti libero di modificare le dimensioni delle risorse e le allocazioni on demand.

Le seguenti domande si concentrano su queste considerazioni relative all'affidabilità. (Per l'elenco completo delle domande e delle best practice relative all'affidabilità, consulta l' Appendice.).

REL 1 In che modo gestisci quote e vincoli di servizio?

Per le architetture di carichi di lavoro basate sul cloud, esistono quote di servizio (definite anche come restrizioni dei servizi). Queste quote sono presenti per evitare di effettuare accidentalmente il provisioning di più risorse di quelle necessarie e limitare i tassi di richiesta sulle operazioni API

REL 1 In che modo gestisci quote e vincoli di servizio?

in modo da proteggere i servizi da un uso illecito. Esistono anche vincoli di risorse, ad esempio la velocità con cui è possibile trasferire i bit su un cavo in fibra ottica o la quantità di storage su un disco fisico.

REL 2 In che modo pianifichi la topologia di rete?

I carichi di lavoro sono spesso presenti in più ambienti. Questi includono più ambienti cloud (sia pubblicamente accessibili sia privati) e, possibilmente, l'infrastruttura del data center esistente. I piani devono includere considerazioni di rete, ad esempio connettività intrasistema e intersistema, gestione di indirizzi IP pubblici, gestione di indirizzi IP privati e risoluzione dei nomi di dominio.

Per le architetture di carichi di lavoro basate sul cloud, esistono quote di servizio (definite anche come restrizioni dei servizi). Queste quote sono presenti per evitare di effettuare accidentalmente il provisioning di più risorse di quelle necessarie e limitare i tassi di richiesta sulle operazioni API in modo da proteggere i servizi da un uso illecito. I carichi di lavoro sono spesso presenti in più ambienti. È necessario monitorare e gestire queste quote per tutti gli ambienti dei carichi di lavoro. Questi includono più ambienti cloud (sia pubblicamente accessibili sia privati) e possono includere l'infrastruttura del data center esistente. I piani devono includere considerazioni di rete, ad esempio connettività intrasistema e intersistema, gestione di indirizzi IP pubblici, gestione di indirizzi IP privati e risoluzione dei nomi di dominio.

Architettura del carico di lavoro

Un carico di lavoro affidabile comincia con decisioni iniziali di progettazione sia per il software sia per l'infrastruttura. Le tue scelte architetturali avranno un impatto sul comportamento del carico di lavoro su tutti e cinque i pilastri del Well-Architected Framework. Per l'affidabilità, è necessario seguire modelli specifici.

Con AWS, gli sviluppatori di carichi di lavoro possono scegliere i linguaggi e le tecnologie da utilizzare. Gli SDK AWS semplificano la scrittura di codici fornendo API specifiche dei linguaggi per i servizi AWS. Questi SDK, oltre alla scelta dei linguaggi, consentono agli sviluppatori di implementare le best practice di affidabilità elencate qui. Gli sviluppatori possono anche leggere e scoprire come Amazon crea e gestisce software nella Amazon Builders' Library.

Le seguenti domande si concentrano su queste considerazioni relative all'affidabilità.

REL 3 In che modo progetti l'architettura del servizio di carico di lavoro?

Creazione di carichi di lavoro altamente scalabili e affidabili utilizzando un'architettura orientata ai servizi (SOA) o un'architettura di microservizi. L'architettura orientata ai servizi (SOA) è la pratica di rendere i componenti software riutilizzabili tramite interfacce di servizio. L'architettura dei microservizi va oltre, per rendere i componenti più piccoli e semplici.

REL 4 In che modo progetti le interazioni in un sistema distribuito per evitare errori?

I sistemi distribuiti si basano sulle reti di comunicazione per interconnettere i componenti (ad esempio server o servizi). Il carico di lavoro deve funzionare in modo affidabile nonostante la perdita o la latenza dei dati in queste reti. I componenti del sistema distribuito devono funzionare in modo da non influire negativamente su altri componenti o sul carico di lavoro. Queste best practice prevengono gli errori e migliorano il tempo medio tra errori (MTBF).

REL 5 In che modo progetti le interazioni in un sistema distribuito per mitigare o affrontare gli errori?

I sistemi distribuiti si basano sulle reti di comunicazione per interconnettere i componenti (ad esempio server o servizi). Il carico di lavoro deve funzionare in modo affidabile nonostante la perdita o la latenza dei dati su queste reti. I componenti del sistema distribuito devono funzionar e in modo da non influire negativamente su altri componenti o sul carico di lavoro. Queste best practice consentono ai carichi di lavoro di affrontare stress o guasti, recuperare più rapidamen te e mitigare l'impatto di tali problemi. Il risultato è un miglioramento del tempo medio di ripristino (MTTR).

Gestione delle modifiche

Le modifiche apportate al carico di lavoro o al relativo ambiente devono essere previste e gestite affinché il carico di lavoro funzioni in modo affidabile. Certe modifiche al carico di lavoro sono imposte da fattori esterni, quali i picchi di domanda, altre modifiche dipendono da fattori interni, quali le distribuzioni delle funzionalità e le patch di sicurezza.

Utilizzando AWS, puoi monitorare il comportamento di un carico di lavoro e automatizzare la risposta ai KPI. Ad esempio, il carico di lavoro può aggiungere ulteriori server man mano che il carico di lavoro acquisisce più utenti. È possibile controllare chi dispone dell'autorizzazione per apportare modifiche al carico di lavoro e controllare la cronologia di tali modifiche.

Le seguenti domande si concentrano su queste considerazioni relative all'affidabilità.

REL 6 In che modo monitori le risorse del carico di lavoro?

I log e i parametri sono strumenti molto efficaci per ottenere informazioni sullo stato del tuo carico di lavoro. È possibile configurare il carico di lavoro in modo da monitorare i log e i parametri e inviare notifiche quando vengono superate le soglie o si verificano eventi significativi. Il monitorag gio consente al carico di lavoro di riconoscere quando vengono superate le soglie di prestazioni basse o si verificano errori, in modo che possa essere ripristinato automaticamente di rimando.

REL 7 In che modo progetti il carico di lavoro per adattarti ai cambiamenti della domanda?

Un carico di lavoro scalabile fornisce elasticità per aggiungere o rimuovere risorse automatic amente, in modo che vi sia una stretta corrispondenza con la domanda attuale in un dato momento.

REL 8 In che modo implementi le modifiche?

Per distribuire nuove funzionalità e garantire che i carichi di lavoro e l'ambiente operativo eseguano software noti e che sia possibile applicare patch o sostituirli in modo prevedibile, sono necessarie modifiche controllate. Se invece non sono controllate, risulta difficile prevederne l'effetto o risolvere eventuali problemi che causano.

Progettando un carico di lavoro in grado di aggiungere e rimuovere automaticamente le risorse in risposta ai cambiamenti della domanda, non solo si aumenta l'affidabilità, ma ci si assicura anche che il successo aziendale non diventi un peso. Con il monitoraggio attivo, il tuo team verrà avvisato automaticamente quando gli indicatori KPI si discostano dalle norme previste. La registrazione automatica delle modifiche al proprio ambiente consente di controllare e identificare rapidamente

le azioni che potrebbero avere influito sull'affidabilità. I controlli sulla gestione delle modifiche assicurano la possibilità di applicare le regole che garantiscono l'affidabilità di cui hai bisogno.

Gestione degli errori

In qualsiasi sistema di ragionevole complessità è previsto che si verifichino errori. L'affidabilità richiede che il carico di lavoro venga a conoscenza degli errori nel momento in cui si verificano e intervenga per evitare conseguenze sulla disponibilità. I carichi di lavoro devono essere in grado di affrontare errori e risolvere automaticamente i problemi.

Con AWS, puoi sfruttare l'automazione per reagire ai dati di monitoraggio. Ad esempio, quando un determinato parametro supera una soglia, è possibile attivare un'azione automatica per risolvere il problema. Inoltre, anziché tentare di diagnosticare e correggere una risorsa guasta che fa parte del tuo ambiente di produzione, puoi sostituirla con una nuova ed eseguire l'analisi sulla risorsa guasta fuori banda. Poiché il cloud consente di creare versioni temporanee di un intero sistema a basso costo, è possibile utilizzare i test automatizzati per verificare i processi di recupero completi.

Le seguenti domande si concentrano su queste considerazioni relative all'affidabilità.

REL 9 In che modo esegui il backup dei dati?

Esegui il backup dei dati, delle applicazioni e della configurazione per soddisfare i tuoi requisiti relativi agli obiettivi di tempo di ripristino (recovery time objective, RTO) e agli obiettivi di punto di ripristino (recovery point objective, RPO).

REL 10 In che modo utilizzi l'isolamento dei guasti per proteggere il carico di lavoro?

Le barriere per l'isolamento dei guasti limitano l'effetto di un errore all'interno di un carico di lavoro a un numero limitato di componenti. I componenti al di fuori della barriera non subiscono gli effetti del guasto. Utilizzando più barriere per l'isolamento dei guasti, puoi limitare l'impatto sul carico di lavoro.

REL 11 In che modo progetti il carico di lavoro affinché resista ai guasti dei componenti?

I carichi di lavoro con requisiti di disponibilità elevata e MTTR (Mean Time To Recovery) basso devono essere progettati per garantire la resilienza.

REL 12 In che modo testi l'affidabilità?

Dopo aver progettato il carico di lavoro in modo da essere resiliente alle sollecitazioni della produzione, i test sono l'unico modo per garantire il funzionamento corretto e offrire la resilienza prevista.

REL 13 Come pianifichi il disaster recovery (DR)?

Avere backup e componenti del carico di lavoro ridondanti in loco è l'inizio della strategia di disaster recovery. RTO e RPO sono i tuoi obiettivi per il ripristino del carico di lavoro. Imposta questi valori in base alle esigenze aziendali. Implementa una strategia per raggiungere questi obiettivi, prendendo in considerazione le posizioni e la funzione delle risorse e dei dati del carico di lavoro. La probabilità di interruzione e il costo del ripristino sono fattori chiave che aiutano a comunicare il valore aziendale che può avere il ripristino di emergenza per un carico di lavoro.

Esegui regolarmente il backup dei dati e testa i file di backup per assicurarti di poter effettuare il ripristino dopo errori sia logici che fisici. Una chiave per la gestione dei guasti è il test frequente e automatico dei carichi di lavoro che causano gli errori e quindi osservare come si ripristinano. Esegui questa operazione regolarmente e assicurati che tali test vengano attivati anche dopo importanti cambiamenti del carico di lavoro. Traccia attivamente i KPI, oltre a Obiettivo del tempo di ripristino (RTO) e Obiettivo del punto di ripristino (RPO), per valutare la resilienza di un carico di lavoro (specialmente in scenari di test degli errori). Il monitoraggio dei KPI ti aiuterà a identificare e mitigare i singoli punti di errore. L'obiettivo è testare a fondo i processi di ripristino del carico di lavoro in modo da avere la certezza di poter recuperare tutti i dati e continuare a servire i propri clienti, anche di fronte a problemi prolungati. I processi di recupero dovrebbero essere testati tanto quanto i normali processi di produzione.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle best practice per l'affidabilità.

Documentazione

- Documentazione di AWS
- Infrastruttura globale di AWS

- AWS Auto Scaling: come funzionano i piani di dimensionamento
- Che cos'è AWS Backup?

Whitepaper

- Pilastro dell'affidabilità: Well-Architected AWS
- Implementazione di microservizi in AWS

Efficienza delle prestazioni

Il principio dell'efficienza delle prestazioni comprende l'abilità di utilizzare in modo efficiente le risorse di elaborazione per soddisfare i requisiti del sistema e conservare tale efficienza a seconda dei cambiamenti della domanda e dell'evoluzione delle tecnologie.

Il principio dell'efficienza delle prestazioni offre una panoramica dei principi di progettazione, delle best practice e delle domande. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul principio dell'efficienza delle prestazioni.

Argomenti

- Principi di progettazione
- Definizione
- Best practice
- Risorse

Principi di progettazione

Esistono cinque principi di progettazione per l'efficienza delle prestazioni nel cloud:

- Estendi a tutti le tecnologie avanzate: facilita l'implementazione di tecnologie avanzate da parte del tuo team delegando le attività complesse al tuo fornitore di cloud. Anziché chiedere al team IT di imparare come adottare e gestire una nuova tecnologia, valuta l'opportunità di utilizzare la tecnologia come servizio. Ad esempio, i database NoSQL, la transcodifica multimediale e il machine learning sono tutte tecnologie che richiedono competenze specialistiche. Nel cloud, tali tecnologie diventano servizi che il tuo team può semplicemente utilizzare mentre si concentra sullo sviluppo di un prodotto invece che sul provisioning e sulla gestione delle risorse.

- Passa a una disponibilità a livello globale in pochi minuti: distribuire il carico di lavoro in più regioni AWS in tutto il mondo ti consente di ridurre la latenza e di fornire un'esperienza migliore ai tuoi clienti a costi minimi.
- Utilizza architetture serverless: scegliendo le architetture serverless, non avrai più bisogno di gestire e mantenere server fisici per portare a termine le attività di elaborazione tradizionali. Ad esempio, i servizi di storage possono agire da siti web statici, eliminando la necessità di server web, mentre i servizi di eventi possono ospitare il codice. Questo elimina l'onere operativo della gestione dei server fisici, con una riduzione dei costi delle transazioni, dal momento che servizi gestiti di questo tipo funzionano a livello di cloud.
- Sperimenta con maggiore frequenza: le risorse virtuali e automatizzabili ti permettono di portare a termine velocemente i test comparativi utilizzando diversi tipi di istanze, storage o configurazioni.
- Considera la comprensione di strumenti e sistemi (mechanical sympathy): scopri come vengono utilizzati i servizi cloud e applica sempre l'approccio tecnologico più adatto ai tuoi obiettivi di carico di lavoro. Ad esempio, prendi in considerazione gli schemi di accesso ai dati quando selezioni una strategia basata su database o archiviazione.

Definizione

Esistono cinque aree di best practice per l'efficienza delle prestazioni nel cloud:

- Scelta dell'architettura
- Calcolo e hardware
- Gestione dati
- Reti e distribuzione di contenuti
- Processo e cultura

Utilizza un approccio basato sui dati per la creazione di un'architettura a prestazioni elevate. Raccogli dati su tutti gli aspetti dell'architettura, dalla progettazione di alto livello alla selezione e alla configurazione dei tipi di risorse.

Rivedendo le tue decisioni a intervalli regolari, avrai la certezza di sfruttare le capacità in continua evoluzione del cloud AWS. Il monitoraggio ti assicurerà di essere consapevole di qualsiasi divergenza rispetto alle prestazioni previste. Infine, puoi raggiungere dei compromessi nella tua architettura per migliorare le prestazioni, per esempio utilizzando la compressione o la memorizzazione nella cache oppure allentando i requisiti di coerenza.

Best practice

Argomenti

Scelta dell'architettura

La soluzione ottimale per un determinato carico di lavoro può variare e le soluzioni spesso combinano molteplici approcci. I carichi di lavoro Well-Architected utilizzano soluzioni multiple e impiegano funzionalità diverse per migliorare le prestazioni.

Le risorse AWS sono disponibili in diverse configurazioni e tipologie, il che semplifica la ricerca di un approccio che soddisfi appieno le tue esigenze. Inoltre, puoi trovare opzioni che non sono facili da trovare nelle infrastrutture in locale. Ad esempio, un servizio gestito come Amazon DynamoDB offre un database NoSQL interamente gestito, con una latenza di pochissimi millisecondi, indipendentemente dalle dimensioni.

Le seguenti domande si concentrano su queste considerazioni relative all'efficienza delle prestazioni. (Per l'elenco completo delle domande e delle best practice relative all'efficienza delle prestazioni, consulta l' Appendix.).

PERF 1: How do you select appropriate cloud resources and architecture patterns for your workload?

Often, multiple approaches are required for more effective performance across a workload. Well-Architected systems use multiple solutions and features to improve performance.

Calcolo e hardware

La soluzione ottimale di elaborazione per un determinato sistema potrebbe variare in base alla progettazione dell'applicazione, ai modelli di utilizzo e alle impostazioni di configurazione. Le architetture possono utilizzare diverse soluzioni di calcolo per vari componenti e impiegare

funzionalità diverse per migliorare le prestazioni. Selezionare la soluzione di elaborazione sbagliata per un'architettura può ridurre l'efficienza delle prestazioni.

In AWS, l'elaborazione è disponibile in tre forme: istanze, container e funzioni.

- Le istanze sono server virtualizzati che consentono di modificare le loro funzionalità con un pulsante o una chiamata API. Poiché nel cloud le decisioni relative alle risorse non sono cristallizzate nel tempo, è possibile sperimentare vari tipi di server. In AWS, tali istanze di server virtuali sono disponibili in famiglie e dimensioni diverse e offrono un'ampia gamma di funzionalità, tra cui unità a stato solido (SSD) e unità di elaborazione grafica (GPU).
- I container sono un metodo di virtualizzazione del sistema operativo che consente di eseguire un'applicazione e le relative dipendenze in processi isolati dalle risorse. Puoi scegliere AWS Fargate, un servizio di elaborazione serverless per container, oppure Amazon EC2, se hai bisogno di controllare l'installazione, la configurazione e la gestione del tuo ambiente di elaborazione. Puoi anche scegliere tra diverse piattaforme di orchestrazione di container: Amazon Elastic Container Service (ECS) o Amazon Elastic Kubernetes Service (EKS).
- Le funzioni astraggono l'ambiente di esecuzione dal codice che si desidera applicare. Ad esempio, AWS Lambda consente di eseguire codice senza eseguire un'istanza.

Le seguenti domande si concentrano su queste considerazioni relative all'efficienza delle prestazioni.

PERF 2: How do you select and use compute resources in your workload?

The more efficient compute solution for a workload varies based on application design, usage patterns, and configuration settings. Architectures can use different compute solutions for various components and turn on different features to improve performance. Selecting the wrong compute solution for an architecture can lead to lower performance efficiency.

Gestione dati

La soluzione ottimale per la gestione dei dati in un sistema specifico varia in base al tipo di dati (blocco, file o oggetto), agli schemi di accesso (casuali o sequenziali), alla velocità di trasmissione effettiva necessaria, alla frequenza di accesso (online, offline, archivio), alla frequenza di aggiornamento (WORM, dinamico) e ai vincoli di disponibilità e durata. I carichi di lavoro Well-Architected utilizzano archivi dati appositamente progettati che impiegano diverse funzionalità per migliorare le prestazioni.

In AWS, lo storage è disponibile in tre forme: oggetto, blocco e file:

- Lo storage di oggetti fornisce una piattaforma scalabile e durevole che rende i dati accessibili da qualsiasi posizione Internet per contenuti generati dagli utenti, archivi attivi, elaborazione serverless, archiviazione di Big Data o backup e ripristino. Amazon Simple Storage Service (Amazon S3) è un servizio di storage di oggetti che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni leader di settore. Amazon S3 è progettato per garantire una durabilità del 99,999999999% (11 nove) e memorizza i dati per milioni di applicazioni per aziende in tutto il mondo.
- Lo storage a blocchi fornisce archiviazione a blocchi a disponibilità elevata, costante e a bassa latenza per ogni host virtuale ed è analogo allo storage collegato direttamente (DAS) o a una rete SAN (Storage Area Network). Amazon Elastic Block Store (Amazon EBS) è stato progettato per carichi di lavoro che richiedono archiviazione persistente accessibile dalle istanze EC2 e consente di ottimizzare le applicazioni con capacità di archiviazione, prestazioni e costi ottimali.
- Lo storage di file fornisce accesso a un file system condiviso tra più sistemi. Le soluzioni di storage di file come Amazon Elastic File System (Amazon EFS) sono ideali per casi d'uso come repository di contenuti di grandi dimensioni, ambienti di sviluppo, store multimediali o home directory. Amazon FSx rende più semplice e conveniente l'avvio e l'esecuzione di file system diffusi in modo da sfruttare le funzionalità avanzate e le prestazioni rapide dei file system open source più utilizzati e con licenza commerciale.

Le seguenti domande si concentrano su queste considerazioni relative all'efficienza delle prestazioni.

PERF 3: How do you store, manage, and access data in your workload?

The more efficient storage solution for a system varies based on the kind of access operation (block, file, or object), patterns of access (random or sequential), required throughput, frequency of access (online, offline, archival), frequency of update (WORM, dynamic), and availability and durability constraints. Well-architected systems use multiple storage solutions and turn on different features to improve performance and use resources efficiently.

Reti e distribuzione di contenuti

La soluzione di rete ottimale per un carico di lavoro varia in base a latenza, requisiti di velocità di trasmissione effettiva, jitter e larghezza di banda. I vincoli fisici, ad esempio le risorse utente o in

locale, determinano le opzioni di posizione. Questi vincoli possono essere compensati con le edge location o la collocazione delle risorse.

In AWS, le reti sono virtualizzate e vengono fornite in molti tipi e configurazioni diversi. In questo modo puoi soddisfare le tue esigenze di rete più facilmente. AWS offre caratteristiche di prodotto (ad esempio reti avanzate, istanze Amazon EC2 ottimizzate per la rete, accelerazione del trasferimento Amazon S3 e Amazon CloudFront dinamico) pensate per l'ottimizzazione del traffico di rete. AWS offre anche funzionalità di rete (ad esempio instradamento in base alla latenza di Amazon Route 53, endpoint Amazon VPC, AWS Direct Connect e AWS Global Accelerator) per ridurre la distanza di rete o il jitter.

Le seguenti domande si concentrano su queste considerazioni relative all'efficienza delle prestazioni.

PERF 4: How do you select and configure networking resources in your workload?

This question includes guidance and best practices to design, configure, and operate efficient networking and content delivery solutions in the cloud.

Processo e cultura

Durante la fase di progettazione dei carichi di lavoro, esistono principi e pratiche che è possibile adottare per gestire al meglio carichi di lavoro cloud efficienti e ad alte prestazioni. Per adottare una cultura che promuova l'efficienza delle prestazioni dei carichi di lavoro cloud, prendi in considerazione questi principi e pratiche fondamentali.

Per sviluppare questa cultura, considera questi principi chiave:

- Infrastruttura come codice: definisci la tua infrastruttura come codice tramite approcci come i modelli di AWS CloudFormation. L'uso dei modelli ti consente di collocare la tua infrastruttura nel controllo sorgente, insieme al codice e alle configurazioni dell'applicazione. Ciò ti premette di applicare le stesse procedure di sviluppo software all'infrastruttura, in modo da accelerare l'iterazione.
- Pipeline di distribuzione: usa una pipeline di integrazione continua/distribuzione continua (CI/CD) (ad esempio repository del codice sorgente, sistemi di sviluppo, distribuzione e automazione dei test) per distribuire la tua infrastruttura. Ciò ti consente di effettuare l'implementazione in modo ripetibile, coerente ed economicamente vantaggioso nel corso dell'iterazione.
- Metriche ben definite: configura e monitora le metriche per raccogliere gli indicatori chiave di prestazione (KPI). Ti consigliamo di adottare parametri tecnici e aziendali. Per i siti Web o le app

mobili, le metriche principali sono il tempo di acquisizione al primo byte o il rendering. Gli altri parametri generalmente validi includono il numero di thread, il tasso di raccolta di dati superflui e gli stati di attesa. I parametri aziendali, come il costo cumulativo aggregato per richiesta, possono indicarti due modi per ridurre i costi. Valuta attentamente il modo in cui prevedi di interpretare i parametri. Ad esempio, potresti scegliere il 99° percentile o quello massimo anziché il valore medio.

- Automatizza i test delle prestazioni: nell'ambito del processo di implementazione, avvia automaticamente i test delle prestazioni dopo che i test di esecuzione più rapidi hanno dato esito positivo. L'automazione deve creare un nuovo ambiente, configurare le condizioni iniziali come i dati del test ed eseguire una serie di benchmark e test di carico. I risultati dei test devono essere confrontati con la build, in modo da monitorare le variazioni delle prestazioni nel corso del tempo. Per i test di lunga durata, puoi inserirli nella pipeline in maniera asincrona rispetto al resto della build. In alternativa, puoi eseguire i test delle prestazioni negli orari notturni, tramite le istanze Spot di Amazon EC2.
- Generazione del carico: crea una serie di script di test che replichino i percorsi utente sintetici o pre-registrati. Tali script devono essere idempotenti e non devono essere associati in coppie. Inoltre, potrebbe essere necessario includere script preliminari per garantire risultati validi. Testa gli script il più possibile, per assicurarti che replichino le abitudini di utilizzo in produzione. Puoi usare soluzioni software o SaaS (Software-as-a-Service) per generare il carico. Valuta se utilizzare le soluzioni Marketplace AWS e le istanze spot: possono essere modi convenienti per generare il carico.
- Visibilità delle prestazioni: i parametri principali devono essere visibili dal team, in particolar modo quelli relativi a ciascuna versione della build. Ciò ti consente di rilevare tendenze positive o negative rilevanti nel corso del tempo. Dovresti anche visualizzare i parametri sul numero di errori o eccezioni per assicurarti di testare un sistema funzionante.
- Visualizzazione: sfrutta le tecniche di visualizzazione che indicano in modo chiaro i punti in cui si verificano problemi di prestazioni, hot spot, stati di attesa o utilizzo ridotto. Sovrapponi i parametri delle prestazioni ai diagrammi architetturali: i grafici delle chiamate o il codice possono aiutarti a individuare più rapidamente i problemi.
- Processo di revisione regolare: le prestazioni scarse delle architetture sono in genere il risultato di un processo di revisione delle prestazioni inesistente o incompleto. Se la tua architettura offre prestazioni insufficienti, l'implementazione di un processo di revisione delle prestazioni ti consente di favorire il miglioramento delle iterazioni.
- Ottimizzazione continua: adotta una cultura per ottimizzare continuamente l'efficienza delle prestazioni del tuo carico di lavoro cloud.

Le seguenti domande si concentrano su queste considerazioni relative all'efficienza delle prestazioni.

PERF 5: What process do you use to support more performance efficiency for your workload?

When architecting workloads, there are principles and practices that you can adopt to help you better run efficient high-performing cloud workloads. To adopt a culture that fosters performance efficiency of cloud workloads, consider these key principles and practices.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle best practice relative all'efficienza delle prestazioni.

Documentazione

- Ottimizzazione delle prestazioni di Amazon S3
- Prestazioni dei volumi di Amazon EBS

Whitepaper

- Il principio dell'efficienza delle prestazioni

Video

- AWS re:Invent 2019: Amazon EC2 foundations (CMP211-R2)
- AWS re:Invent 2019: Leadership session: Storage state of the union (STG201-L)
- AWS re:Invent 2019: Leadership session: AWS purpose-built databases (DAT209-L)
- AWS re:Invent 2019: Connectivity to AWS and hybrid AWS network architectures (NET317-R1)
- AWS re:Invent 2019: Powering next-gen Amazon EC2: Deep dive into the Nitro system (CMP303-R2)
- AWS re:Invent 2019: Scaling up to your first 10 million users (ARC211-R)

Ottimizzazione dei costi

Il principio dell'ottimizzazione dei costi include la possibilità di eseguire sistemi per offrire valore aggiunto al prezzo più basso.

Il principio dell'ottimizzazione dei costi offre una panoramica dei principi di progettazione, delle best practice e delle domande. È possibile trovare linee guida prescrittive sull'implementazione nel Whitepaper sul principio dell'ottimizzazione dei costi.

Argomenti

Principi di progettazione

I principi di progettazione per l'ottimizzazione dei costi nel cloud sono cinque:

- Implementa la gestione finanziaria del cloud: per migliorare i risultati finanziari e accelerare la realizzazione del valore aziendale nel cloud, devi investire nella gestione finanziaria e nell'ottimizzazione dei costi sul cloud. L'organizzazione deve dedicare tempo e risorse per creare capacità in questo nuovo dominio di gestione della tecnologia e dell'utilizzo. Come per le tue funzionalità di sicurezza o eccellenza operativa, devi creare capacità tramite lo sviluppo di competenze, programmi, risorse e processi, per diventare un'organizzazione efficiente in termini di costi.
- Adotta un modello a consumo: paga solo le risorse di calcolo che richiedi e incrementa o riduci l'utilizzo a seconda dei requisiti aziendali, e non attraverso il ricorso a una previsione elaborata. Ad esempio, gli ambienti di test e di sviluppo sono generalmente usati solo per otto ore al giorno durante la settimana lavorativa. Puoi interrompere queste risorse quando non le utilizzi, risparmiando potenzialmente il 75% dei costi (40 ore anziché 168).
- Misura l'efficienza complessiva: misura il risultato aziendale del carico di lavoro e i costi associati alla sua produzione. Usa questi dati per conoscere i ricavi che ottieni grazie all'aumento della produttività e alla riduzione dei costi.
- Smetti di spendere denaro per onerose attività indifferenziate: AWS si occupa delle attività onerose dei data center come il racking, lo stacking e l'alimentazione dei server. Inoltre, elimina l'onere

operativo della gestione di sistemi operativi e applicazioni con servizi gestiti. In questo modo, potrai concentrarti sui tuoi clienti e sui progetti aziendali anziché sull'infrastruttura IT.

- Analizza e attribuisci la spesa: il cloud ti aiuta a individuare con facilità e precisione l'utilizzo e il costo dei sistemi, il che consente quindi l'attribuzione trasparente dei costi IT per i singoli proprietari del carico di lavoro. Questo ti aiuta a misurare il ritorno sull'investimento (ROI) e offre ai proprietari del carico di lavoro la possibilità di ottimizzare le proprie risorse e ridurre i costi.

Definizione

Esistono cinque aree di best practice per l'ottimizzazione dei costi nel cloud:

- Implementazione della gestione finanziaria del cloud
- Consapevolezza delle spese e dell'utilizzo
- Risorse convenienti
- Gestione delle risorse di domanda e offerta
- Ottimizzazione nel tempo

Come per gli altri principi di base all'interno del Canone di architettura, occorre considerare alcuni compromessi; ad esempio, è meglio ottimizzare la velocità di commercializzazione o i costi? In alcuni casi, è meglio ottimizzare la velocità: entrare nel mercato rapidamente, distribuire nuove caratteristiche o semplicemente rispettare una scadenza piuttosto che investire nell'ottimizzazione anticipata dei costi. Talvolta le decisioni di progettazione sono guidate dalla rapidità invece che dai dati, ed esiste sempre la tentazione di sovrascrivere piuttosto che dedicare tempo all'esecuzione di benchmark per la distribuzione più conveniente. Questo potrebbe portare a distribuzioni sovra-assegnate e sotto-ottimizzate. Tuttavia, si tratta di una scelta ragionevole quando devi trasferire le risorse dal tuo ambiente locale al cloud ed eseguire l'ottimizzazione di conseguenza. Investire in anticipo la giusta quantità di energia in una strategia di ottimizzazione dei costi consente di realizzare i vantaggi economici del cloud in modo più rapido, assicurando il rispetto costante delle best practice ed evitando un provisioning superfluo. Le sezioni seguenti forniscono tecniche e best practice per l'implementazione iniziale e continua della gestione finanziaria del cloud e l'ottimizzazione dei costi dei carichi di lavoro.

Best practice

Argomenti

- Implementazione della gestione finanziaria del cloud

- Consapevolezza delle spese e dell'utilizzo
- Risorse convenienti
- Gestione delle risorse di domanda e offerta
- Ottimizzazione nel tempo

Implementazione della gestione finanziaria del cloud

Con l'adozione del cloud, i team tecnologici innovano più rapidamente grazie a cicli di approvazione, approvvigionamento e distribuzione dell'infrastruttura più brevi. Per ottenere valore aggiunto e migliorare gli affari è necessario un nuovo approccio alla gestione finanziaria nel cloud. Questo approccio è la gestione finanziaria del cloud e crea capacità in tutta l'organizzazione implementando competenze, programmi, risorse e processi a livello organizzativo.

Molte organizzazioni sono composte da tante unità con priorità diverse. La capacità di allineare un'organizzazione a un insieme concordato di obiettivi finanziari e di fornire all'organizzazione i meccanismi per raggiungerli permette di creare un'organizzazione più efficiente. Un'organizzazione capace innova e crea più rapidamente, è più agile e si adatta a qualsiasi fattore interno o esterno.

In AWS puoi utilizzare Cost Explorer e, facoltativamente, Amazon Athena e Amazon QuickSight con il report costi e utilizzo (CUR) per fornire consapevolezza su costi e utilizzo in tutta l'organizzazione. Budget AWS fornisce notifiche proattive relative a costi e utilizzo. I blog AWS forniscono informazioni su nuovi servizi e caratteristiche per consentirti di essere sempre aggiornato sulle nuove versioni dei servizi.

La seguente domanda si concentra su queste considerazioni relative all'ottimizzazione dei costi. (Per l'elenco completo delle domande e delle best practice relative all'ottimizzazione dei costi, consulta l' Appendice.).

COST 1 In che modo implementi la gestione finanziaria nel cloud?

L'implementazione della gestione finanziaria del cloud consente alle organizzazioni di conseguire un valore aggiunto e il successo finanziario ottimizzando i costi e l'utilizzo e ricalibrando le risorse in AWS.

Quando crei una funzione di ottimizzazione dei costi, puoi utilizzare i membri e integrare il team con esperti di gestione finanziaria del cloud e ottimizzazione dei costi. I membri già presenti nel

team conoscono il funzionamento dell'organizzazione e sono in grado di implementare rapidamente i miglioramenti. Valuta anche la possibilità di includere persone con competenze aggiuntive o specialistiche, ad esempio di analisi e gestione dei progetti.

Quando implementi la consapevolezza dei costi nella tua organizzazione, prova a migliorare o sviluppare i programmi e i processi esistenti. È molto più veloce sviluppare i processi e programmi esistenti, piuttosto che crearne di nuovi. In questo modo puoi ottenere risultati molto più rapidamente.

Consapevolezza delle spese e dell'utilizzo

La maggiore flessibilità e agilità consentite dal cloud incoraggiano l'innovazione, lo sviluppo e la distribuzione rapidi. Elimina i processi manuali e il tempo associati al provisioning dell'infrastruttura locale, tra cui l'identificazione delle specifiche hardware, la negoziazione delle quotazioni dei prezzi, la gestione degli ordini di acquisto, la pianificazione delle spedizioni e la distribuzione delle risorse. Tuttavia, la facilità d'uso e la capacità on demand virtualmente illimitata richiedono un nuovo tipo di mentalità in merito alle spese.

Molte aziende sono caratterizzate da più sistemi gestiti da vari team. La capacità di attribuire i costi delle risorse ai singoli proprietari dell'organizzazione o del prodotto incoraggia un comportamento di utilizzo efficiente e contribuisce a ridurre gli sprechi. L'attribuzione precisa dei costi consente di capire quali prodotti sono effettivamente redditizi e permette anche di prendere decisioni più consapevoli in merito alle destinazioni del budget.

Con AWS puoi creare una struttura di account con AWS Organizations o AWS Control Tower per garantire la separazione e semplificare l'allocazione di costi e utilizzo. Puoi anche utilizzare l'applicazione di tag alle risorse per associare informazioni aziendali e organizzative a utilizzo e costi. Utilizza AWS Cost Explorer per osservare costi e utilizzo, oppure crea analisi e pannelli di controllo personalizzati con Amazon Athena e Amazon QuickSight. Puoi verificare costi e utilizzo con le notifiche di Budget AWS e controllarli usando AWS Identity and Access Management (IAM) e Service Quotas.

Le seguenti domande si concentrano su queste considerazioni relative all'ottimizzazione dei costi.

COST 2 In che modo gestisci l'utilizzo?

Stabilisci policy e meccanismi per assicurarti di sostenere costi adeguati mentre raggiungi gli obiettivi. Utilizzando un approccio di controllo e bilanciamento reciproco, è possibile innovare senza spendere troppo.

COST 3 In che modo monitori l'utilizzo e il costo?

Stabilisci policy e procedure per monitorare e allocare i costi in modo appropriato. Ciò ti consente di misurare e migliorare l'efficienza in termini di costi del carico di lavoro.

COST 4 In che modo ritiri le risorse?

Implementa il controllo del cambiamento e la gestione delle risorse dall'inizio del progetto alla fine del ciclo di vita. In questo modo, puoi chiudere o interrompere le risorse non utilizzate per ridurre gli sprechi.

Puoi usare i tag di allocazione dei costi per categorizzare e monitorare il tuo utilizzo di AWS e i costi. Quando applichi dei tag alle tue risorse AWS (come le istanze EC2 o i bucket S3), AWS genera un report su costi e utilizzo con i tuoi tag e i dati sul tuo utilizzo. Puoi applicare tag che rappresentano le categorie di un'organizzazione (come i centri di costo, i nomi dei carichi di lavoro o i proprietari) per organizzare i tuoi costi tra i vari servizi.

Assicurati di utilizzare il giusto livello di dettaglio e granularità quando crei report e monitori costi e utilizzo. Per informazioni e tendenze generali, utilizza i dati giornalieri di AWS Cost Explorer. Per analisi e ispezioni più specifiche, utilizza i dati orari di AWS Cost Explorer, oppure Amazon Athena e Amazon QuickSight impostando un livello di granularità oraria nel Report costi e utilizzo.

Associando le risorse taggate al monitoraggio del ciclo di vita dell'entità (dipendenti, progetti), puoi individuare le risorse accantonate o i progetti che non generano più valore per l'organizzazione e devono quindi essere dismessi. Puoi impostare avvisi di fatturazione per ricevere notifiche relative a spese eccessive previste.

Risorse convenienti

Utilizzare risorse e istanze adeguate al tuo carico di lavoro è fondamentale per ridurre i costi. Ad esempio, un processo di reporting potrebbe impiegare cinque ore su un server più piccolo, ma un'ora su un server più grande che costa il doppio. Entrambi i server ti offrono lo stesso risultato, ma quello più piccolo comporta un costo più elevato nel tempo.

Un carico di lavoro basato sul Canone di architettura AWS si basa sulle risorse più convenienti, il che può avere un impatto economico positivo e notevole. Hai anche la possibilità di usare i servizi gestiti

per ridurre i costi. Ad esempio, invece di mantenere dei server per recapitare le e-mail, puoi usare un servizio che ti invia gli addebiti in base ai messaggi inviati.

AWS offre un'ampia gamma di offerte flessibili e convenienti per acquisire istanze da Amazon EC2 e altri servizi per soddisfare al meglio le tue necessità. On demand Istanze on demand ti consentono di pagare la capacità di elaborazione a ore e non richiedono impegni minimi. Savings Plans e istanze riservate offrono risparmi fino al 75% rispetto ai prezzi on demand. Con le istanze Spot, puoi sfruttare la capacità inutilizzata di Amazon EC2 e risparmiare fino al 90% sui prezzi on demand. Istanze Spot risultano adeguate quando il sistema può tollerare l'utilizzo di un parco server in cui i singoli server possano andare e venire dinamicamente, come server Web stateless, elaborazioni batch o quando si usano HPC e Big Data.

Anche la scelta del servizio appropriato può ridurre l'utilizzo e i costi; ad esempio, CloudFront può ridurre al minimo il trasferimento dei dati o eliminare del tutto i costi, mentre l'utilizzo di Amazon Aurora su RDS può rimuovere gli elevati costi di licenza dei database.

Le seguenti domande si concentrano su queste considerazioni relative all'ottimizzazione dei costi.

COST 5 In che modo valuti i costi quando selezioni i servizi?

Amazon EC2, Amazon EBS e Amazon S3 sono servizi AWS di base. I servizi gestiti, come Amazon RDS e Amazon DynamoDB, sono servizi AWS di livello superiore o applicativo. Selezionando i blocchi predefiniti e i servizi gestiti appropriati, è possibile ottimizzare questo carico di lavoro per i costi. Ad esempio, utilizzando i servizi gestiti, puoi ridurre o eliminare gran parte dei costi generali amministrativi e operativi, liberandotene per lavorare su applicazioni e attività correlate al tuo business.

COST 6 In che modo raggiungi gli obiettivi di costo quando selezioni il tipo, le dimensioni e il numero delle risorse?

Assicurati di scegliere la dimensione e il numero delle risorse appropriati per l'attività in questione. Selezionando il tipo, le dimensioni e il numero più convenienti, riduci al minimo gli sprechi.

COST 7 In che modo impieghi i modelli di prezzo per ridurre i costi?

Utilizza il modello di prezzo più appropriato per le tue risorse per ridurre al minimo le spese.

COST 8 In che modo pianifichi i costi per il trasferimento dei dati?

Assicurati di pianificare e monitorare i costi di trasferimento dei dati in modo da poter prendere decisioni sull'architettura per ridurre al minimo i costi. Una modifica piccola ma efficace dell'arch itettura può ridurre drasticamente i costi operativi nel tempo.

Scomponendo i costi durante la selezione del servizio e usando strumenti come Cost Explorer e AWS Trusted Advisor per esaminare con regolarità l'utilizzo di AWS, puoi monitorare attivamente il tuo utilizzo e modificare le implementazioni di conseguenza.

Gestione delle risorse di domanda e offerta

Quando passi al cloud, paghi solo ciò che ti occorre. Puoi fornire risorse in base alla domanda del carico di lavoro nel momento in cui sono necessarie, eliminando così la necessità di un provisioning superfluo costoso e dispendioso. Puoi anche gestire la domanda utilizzando tecniche come throttling, buffering o queuing per allentare la domanda e soddisfarla con meno risorse. In questo modo diminuirai i costi o li posticiperai con un servizio batch.

In AWS puoi predisporre automaticamente le risorse da associare alla domanda di carico di lavoro. Auto Scaling con strategie basate su domanda o tempo ti consente di aggiungere e rimuovere le risorse in base alle esigenze. Se riesci a prevedere le variazioni nella domanda, puoi risparmiare di più e assicurarti che le risorse corrispondano alle esigenze del tuo carico di lavoro. Puoi utilizzare Amazon API Gateway per implementare il throttling o Amazon SQS per implementare una coda nel carico di lavoro. Entrambi consentono di modificare la richiesta nei componenti del carico di lavoro.

La seguente domanda si concentra su queste considerazioni relative all'ottimizzazione dei costi.

COST 9 Come gestisci la domanda e fornisci le risorse?

Per avere un carico di lavoro con costo e prestazioni bilanciate, assicurati che venga utilizzato tutto ciò per cui paghi ed evita le istanze molto sottoutilizzate. Un parametro di utilizzo distorto, in qualsiasi delle suddette direzioni, ha un impatto negativo sull'organizzazione, sia per i costi operativi (basse prestazioni a causa di un utilizzo eccessivo) che per le spese AWS sprecate (a causa di un provisioning eccessivo).

Quando progetti di modificare le risorse di domanda e offerta, pensa attentamente ai modelli di utilizzo, al tempo necessario per effettuare il provisioning delle nuove risorse e alla prevedibilità del

modello di domanda. Quando gestisci la domanda, assicurati di disporre di una coda o di un buffer di dimensioni corrette e di rispondere alla domanda del carico di lavoro nel periodo di tempo richiesto.

Ottimizzazione nel tempo

Poiché AWS rilascia nuovi servizi e caratteristiche, è buona prassi rivedere le decisioni correnti sull'architettura per garantire che continuino a essere le più convenienti. Man mano che le tue esigenze cambiano, disattiva tempestivamente risorse, interi servizi e sistemi non appena smettono di essere necessari.

L'implementazione di nuove caratteristiche o tipi di risorse può ottimizzare il carico di lavoro in modo incrementale e con uno sforzo minimo. In questo modo puoi migliorare continuamente l'efficienza nel tempo e essere sicuro di utilizzare le tecnologie più aggiornate per ridurre i costi operativi. Puoi anche sostituire o aggiungere nuovi componenti al carico di lavoro con nuovi servizi. In questo modo puoi aumentare in modo significativo l'efficienza, perciò è essenziale rivedere regolarmente il carico di lavoro e implementare nuovi servizi e caratteristiche.

La seguente domanda si concentra su queste considerazioni relative all'ottimizzazione dei costi.

COST 10 In che modo valuti i nuovi servizi?

Poiché AWS rilascia nuovi servizi e caratteristiche, è buona prassi rivedere le decisioni correnti sull'architettura per garantire che continuino a essere le più convenienti.

Quando esamini regolarmente le tue distribuzioni, valuta in che modo i servizi più recenti possono aiutarti a risparmiare. Ad esempio, Amazon Aurora su RDS può ridurre i costi dei database relazionali. L'utilizzo di serverless come Lambda consente di eliminare la necessità di utilizzare e gestire le istanze per eseguire il codice.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle nostre best practice per l'ottimizzazione dei costi.

Documentazione

- Documentazione di AWS

Whitepaper

- Principio dell'ottimizzazione dei costi

Sostenibilità

Alla base del concetto di Sostenibilità c'è l'attenzione all'impatto ambientale, soprattutto in termini di uso ed efficienza delle fonti energetiche, leve importanti che gli architetti usano per definire interventi diretti mirati a ridurre lo sfruttamento delle risorse. È possibile trovare linee guida prescrittive sull'implementazione nel Whitepaper sul principio della sostenibilità.

Argomenti

- Principi di progettazione
- Definizione
- Best practice

Principi di progettazione

Esistono sei principi di progettazione per la sostenibilità nel cloud:

- Comprendi il tuo impatto: Misura l'impatto del tuo carico di lavoro cloud e definisci il suo impatto futuro. Nella tua analisi includi ogni fonte di impatto: quelle derivanti dall'uso dei prodotti da parte dei tuoi clienti e quelle derivanti dalla rimozione e dal ritiro finali dal mercato. Confronta l'output di produzione e l'impatto totale dei tuoi carichi di lavoro cloud, partendo dall'analisi di risorse ed emissioni richieste per unità di lavoro. Usa questi dati per definire indicatori chiave di prestazione (KPI), capire come migliorare la produttività, riducendo al tempo stesso l'impatto, e stimare l'impatto delle modifiche proposte nel tempo.
- Stabilisci obiettivi di sostenibilità: Per ogni carico di lavoro cloud stabilisci obiettivi di sostenibilità a lungo termine, come, ad esempio, ridurre le risorse di calcolo e di archiviazione richieste per ciascuna transazione. Modella il ritorno sugli investimenti finalizzati alle migliorie in materia di sostenibilità per i carichi di lavoro esistenti e offri ai proprietari le risorse di cui hanno bisogno per investire negli obiettivi di sostenibilità. Pianifica lo sviluppo e progetta i tuoi carichi di lavoro in modo che la crescita comporti un impatto meno intenso se misurato rispetto a un'unità appropriata, come l'utente o la transazione. Gli obiettivi ti aiutano ad avvalorare un progetto più ampio di sostenibilità che coinvolge la tua azienda o la tua organizzazione, a identificare le regressioni e a dare la priorità a quelle aree che offrono un maggiore potenziale di miglioramento.

- Ottimizza l'utilizzo: Dimensiona correttamente i carichi di lavoro e implementa un progetto efficiente in grado di garantire un utilizzo elevato e ottimizzare l'efficienza energetica dell'hardware sottostante. Due host in esecuzione con una percentuale di utilizzo pari al 30% sono meno efficienti di un host in esecuzione al 60%, se consideriamo il consumo di base per host. Allo stesso tempo, elimina o riduci le risorse, le elaborazioni e le archiviazioni inattive per ridurre l'energia totale richiesta per alimentare il tuo carico di lavoro.
- Anticipa e adotta offerte hardware e software nuove e più efficienti: Promuovi le migliorie a monte di partner e fornitori finalizzate a ridurre l'impatto dei carichi di lavoro cloud. Monitora costantemente il mercato e valuta nuove offerte hardware e software più efficienti. Adotta la flessibilità nei tuoi progetti per consentire una rapida adozione di tecnologie nuove ed efficienti.
- Utilizza servizi gestiti: La condivisione dei servizi con un'ampia base clienti consente di ottimizzare l'uso delle risorse e ridurre al tempo stesso l'infrastruttura necessaria per supportare i carichi di lavoro nel cloud. I clienti possono ad esempio condividere l'impatto di componenti comuni di data center, come reti ed energia, migrando i carichi di lavoro su Cloud AWS e adottando servizi gestiti, come AWS Fargate per i container serverless, in cui AWS opera su vasta scala ed è responsabile della loro efficienza operativa. Utilizza i servizi gestiti per contribuire alla riduzione dell'impatto, trasferendo automaticamente dati con accesso poco frequente all'archiviazione dei dati inattivi con le configurazioni di Amazon S3 Lifecycle o di Amazon EC2 Auto Scaling per adeguare le capacità alla domanda.
- Riduci l'impatto a valle dei carichi di lavoro cloud: Diminuisci la quantità di energia o di risorse richieste per l'utilizzo dei tuoi servizi. Riduci o elimina la necessità di eseguire upgrade dei dispositivi per consentire ai clienti di usare i tuoi servizi. Esegui test usando device farm per analizzare l'impatto atteso e conduci altri test con i clienti per capire l'impatto reale derivante dall'uso dei tuoi servizi.

Definizione

Esistono sei aree di best practice per la sostenibilità nel cloud:

- Selezione delle regioni
- Modelli di comportamento degli utenti
- Modelli di software e architetture
- Modelli di dati
- Modelli hardware
- Processo di sviluppo e implementazione

Sostenibilità nel cloud significa impegnarsi continuamente per ridurre principalmente il consumo di energia e garantire una maggiore efficienza di tutti i componenti di un carico di lavoro, ottenendo il massimo vantaggio dalle risorse fornite e riducendo al minimo le quantità richieste. Tale impegno va dalla selezione iniziale di un linguaggio di programmazione efficace, dall'adozione di algoritmi moderni e dall'uso di tecniche di archiviazione di dati efficienti alla distribuzione in infrastrutture di calcolo valide e correttamente dimensionate e alla riduzione dei requisiti per l'hardware degli utenti finali a potenza elevata.

Best practice

Argomenti

Selezione delle regioni

Scegli le Regioni in cui implementerai i tuoi carichi di lavoro, tenendo presenti sia i requisiti aziendali sia gli obiettivi di sostenibilità.

La seguente domanda si concentra su queste considerazioni relative alla sostenibilità. (Per l'elenco completo delle domande e delle best practice relative all'affidabilità, consulta l' Appendice.)

SUS 1: In che modo selezioni le Regioni per sostenere i tuoi obiettivi di sostenibilità?
Scegli le Regioni vicino ai progetti di energia rinnovabile di Amazon e le Regioni in cui la griglia presenta un'intensità di emissione di anidride carbonica nota inferiore a quella di altre sedi (o Regioni).

Modelli di comportamento degli utenti

Il modo in cui gli utenti utilizzano i tuoi carichi di lavoro e altre risorse può aiutarti a identificare i miglioramenti da implementare per raggiungere gli obiettivi di sostenibilità. Dimensiona l'infrastruttura in modo che si adegui continuamente al carico degli utenti e implementa solo le risorse minime richieste per supportare gli utenti. Allinea i livelli di servizio alle esigenze dei clienti. Posiziona le risorse in modo da limitare la rete richiesta per il consumo da parte degli utenti. Elimina risorse esistenti non utilizzate. Identifica le risorse create non utilizzate e smetti di generarle. Offri ai membri del tuo team dispositivi in grado di soddisfare le loro esigenze con un impatto ridotto in termini di sostenibilità.

La seguente domanda si concentra su queste considerazioni relative alla sostenibilità:

SUS 2: In che modo sfrutti i modelli di comportamento degli utenti per sostenere i tuoi obiettivi di sostenibilità?

Il modo in cui gli utenti utilizzano i tuoi carichi di lavoro e altre risorse può aiutarti a identific are i miglioramenti da implementare per raggiungere gli obiettivi di sostenibilità. Dimensiona l'infrastruttura in modo che si adegui continuamente al carico degli utenti e implementa solo le risorse minime richieste per supportare gli utenti. Allinea i livelli di servizio alle esigenze dei clienti. Posiziona le risorse in modo da limitare la rete richiesta per il consumo da parte degli utenti. Elimina risorse esistenti non utilizzate. Identifica le risorse create non utilizzate e smetti di generarle. Offri ai membri del tuo team dispositivi in grado di soddisfare le loro esigenze con un impatto ridotto in termini di sostenibilità.

Dimensiona l'infrastruttura in base al carico degli utenti: identifica i periodi di utilizzo assente o ridotto e riduci le risorse per evitare capacità in eccesso e migliorare il livello di efficienza.

Allinea gli SLA agli obiettivi di sostenibilità: definisci e aggiorna gli Accordi sul Livello di Servizio (SLA), come la disponibilità di periodi di conservazione dei dati, per ridurre il numero di risorse richieste a supporto dei carichi di lavoro, senza per questo venire meno ai requisiti di business.

Elimina la creazione e la manutenzione di asset inutilizzati: analizza le risorse delle applicazioni (come report precompilati, set di dati e immagini statiche) e i modelli di accesso alle risorse per identificare ridondanze, sottoutilizzi e obiettivi potenziali di disattivazione. Consolida le risorse generate con contenuti ridondanti (come, ad esempio, report mensili con set di dati e output comuni o in sovrapposizione) per eliminare le risorse utilizzate per la duplicazione degli output. Disattiva

le risorse non utilizzate (come, ad esempio, immagini di prodotto non più in vendita) per liberare le risorse usate e ridurre il numero di risorse sfruttate per supportare il carico di lavoro.

Ottimizza il posizionamento geografico dei carichi di lavoro in base alle posizioni degli utenti: analizza i modelli di accesso alla rete per capire da quali aree geografiche si connettono i tuoi clienti. Seleziona le Regioni e i servizi per ridurre la distanza che il traffico di rete deve percorrere e diminuire così le risorse totali di rete richieste per supportare il tuo carico di lavoro.

Ottimizza le risorse dei membri del team in base alle attività eseguite: ottimizza le risorse fornite ai membri del team per ridurre al minimo l'impatto sulla sostenibilità e supportare al tempo stesso le loro esigenze. Esegui ad esempio operazioni complesse, come rendering e compilazione, su desktop cloud condivisi altamente utilizzati invece che su sistemi per utenti singoli, sottoutilizzati e con un alto dispendio energetico.

Modelli di software e architetture

Implementa modelli per eseguire lo smoothing del carico e garantire un utilizzo elevato e coerente delle risorse implementate per ridurre al minimo il loro consumo. In seguito alle modifiche nei comportamenti degli utenti nel tempo, alcuni componenti potrebbero diventare inattivi per mancanza di utilizzo. Rivedi modelli e architetture per consolidare i componenti sottoutilizzati e aumentare l'uso complessivo. Ritira i componenti che non sono più necessari. Analizza le prestazioni dei componenti dei tuoi carichi di lavoro e ottimizza quelli che usano la maggior quantità di risorse. Identifica i dispositivi che i clienti utilizzano per accedere ai servizi e implementa modelli in grado di ridurre al minimo la necessità di aggiornamenti dei dispositivi.

Le seguenti domande si concentrano su queste considerazioni relative alla sostenibilità:

SUS 3: In che modo sfrutti i modelli di software e architetture per sostenere i tuoi obiettivi di sostenibilità?

Implementa modelli per eseguire lo smoothing del carico e garantire un utilizzo elevato e coerente delle risorse implementate per ridurre al minimo il loro consumo. In seguito alle modifiche nei comportamenti degli utenti nel tempo, alcuni componenti potrebbero diventare inattivi per mancanza di utilizzo. Rivedi modelli e architetture per consolidare i componenti sottoutilizzati e aumentare l'uso complessivo. Ritira i componenti che non sono più necessari. Analizza le prestazioni dei componenti dei tuoi carichi di lavoro e ottimizza quelli che usano la maggior quantità di risorse. Identifica i dispositivi che i clienti utilizzano per accedere ai servizi e implement a modelli in grado di ridurre al minimo la necessità di aggiornamenti dei dispositivi.

Ottimizza software e architetture per processi asincroni e pianificati: utilizza progettazioni e architetture software efficienti per ridurre al minimo le risorse medie richieste per unità di lavoro. Implementa meccanismi che generano un utilizzo uniforme dei componenti per ridurre le risorse inattive tra le attività e diminuire l'impatto di picchi di carico.

Rimuovi o rifattorizza i componenti dei carichi di lavoro con un utilizzo ridotto o assente: monitora l'attività dei carichi di lavoro per individuare i cambiamenti che si verificano nel tempo nell'utilizzo dei singoli componenti. Elimina i componenti non utilizzati e non più necessari e rifattorizza quelli con scarso utilizzo per limitare lo spreco di risorse.

Ottimizza le aree di codice che consumano la maggior parte del tempo o delle risorse: monitora l'attività dei carichi di lavoro per individuare i componenti delle applicazioni che usano la maggior parte delle risorse. Ottimizza il codice eseguito all'interno di questi componenti per ridurre l'utilizzo delle risorse e massimizzare al tempo stesso le prestazioni.

Ottimizza l'impatto su dispositivi e apparecchiature dei clienti: identifica i dispositivi e le attrezzature che i tuoi clienti usano per accedere ai tuoi servizi, il loro ciclo di vita atteso e l'impatto finanziario e di sostenibilità che deriva dalla loro sostituzione. Implementa modelli e architetture software per ridurre al minimo la necessità dei clienti di sostituire dispositivi e aggiornare attrezzature. Implementa ad esempio nuove caratteristiche usando un codice compatibile con versioni di hardware e sistemi operativi precedenti o gestisci la dimensione dei payload in modo che non superino la capacità di archiviazione del dispositivo target.

Usa i modelli e le architetture software che meglio supportano l'accesso ai dati e i modelli di archiviazione: scopri come i dati vengono utilizzati all'interno del tuo carico di lavoro, consumati dagli utenti, trasferiti e archiviati. Seleziona tecnologie che ti consentono di ridurre l'elaborazione dei dati e i requisiti di archiviazione.

Modelli di dati

Implementa modelli per eseguire lo smoothing del carico e garantire un utilizzo elevato e coerente delle risorse implementate per ridurre al minimo il loro consumo. In seguito alle modifiche nei comportamenti degli utenti nel tempo, alcuni componenti potrebbero diventare inattivi per mancanza di utilizzo. Rivedi modelli e architetture per consolidare i componenti sottoutilizzati e aumentare l'uso complessivo. Ritira i componenti che non sono più necessari. Analizza le prestazioni dei componenti dei tuoi carichi di lavoro e ottimizza quelli che usano la maggior quantità di risorse. Identifica i dispositivi che i clienti utilizzano per accedere ai servizi e implementa modelli in grado di ridurre al minimo la necessità di aggiornamenti dei dispositivi.

La seguente domanda si concentra su queste considerazioni relative alla sostenibilità:

SUS 4: In che modo sfrutti i modelli di accesso e di utilizzo dei dati per sostenere i tuoi obiettivi di sostenibilità?

Implementa procedure di gestione dei dati per ridurre l'archiviazione assegnata richiesta per supportare il carico di lavoro e le risorse necessarie per l'uso correlato. Analizza i tuoi dati e usa tecnologie e configurazioni di archiviazione che meglio supportano il valore aziendale dei dati e il modo in cui vengono utilizzati. Esegui il ciclo di vita dei dati su un'archiviazione più efficiente e meno performante al diminuire dei requisiti ed elimina i dati che non sono più necessari.

Implementa una policy di classificazione dei dati: classifica i dati per comprenderne il significato in favore dei risultati aziendali. Usa queste informazioni per stabilire quando trasferire i dati in un'archiviazione più efficiente dal punto di vista energetico o eliminarli in totale sicurezza.

Utilizza tecnologie che supportano l'accesso ai dati e i modelli di archiviazione: usa l'archiviazione in grado di supportare al meglio il modo in cui viene effettuato l'accesso ai dati e come vengono archiviati per ridurre la quantità di risorse assegnate e supportare al tempo stesso il tuo carico di lavoro. I dispositivi allo stato solido (SSD) utilizzano ad esempio l'energia in modo più intensivo rispetto ai drive magnetici e dovrebbero essere usati solo per casi d'uso di dati attivi. Usa storage di classe di archiviazione ad alta efficienza energetica per i dati ad accesso infrequente.

Utilizza le policy del ciclo di vita per eliminare i dati non necessari: gestisci il ciclo di vita di tutti i tuoi dati e applica in automatico cronologie di eliminazione per ridurre i requisiti totali di archiviazione del tuo carico di lavoro.

Riduci il provisioning eccessivo nell'archiviazione a blocchi: per ridurre la quantità totale di archiviazione assegnata, crea un'archiviazione a blocchi con l'allocazione di dimensioni in base al carico di lavoro. Usa i volumi elastici per espandere l'archiviazione all'aumentare dei dati senza dover ridimensionare l'archiviazione collegata alle risorse di calcolo. Esamina regolarmente i volumi elastici e riduci i volumi con un provisioning eccessivo per adattarli alla dimensione corrente dei dati.

Elimina i dati ridondanti o non necessari: duplica i dati solo quando è necessario per ridurre la quantità totale di archiviazione utilizzata. Utilizza tecnologie di backup che deduplicano i dati a livello di file e blocco. Limita l'uso di configurazioni Redundant Array of Independent Drives (RAID), ad eccezione dei casi in cui sono richieste per soddisfare gli SLA.

Utilizza file system condivisi o archiviazione di oggetti per accedere a dati comuni: adotta l'archiviazione condivisa e singole fonti di verità per evitare la duplicazione dei dati e ridurre i requisiti di archiviazione complessiva del tuo carico di lavoro. Recupera i dati dall'archiviazione condivisa

solo in base alle esigenze. Distacca volumi non utilizzati per liberare le risorse. Riduci al minimo gli spostamenti dei dati tra le reti: usa un'archiviazione condivisa e accedi ai dati da archivi regionali per contenere le risorse di rete totali necessarie per supportare i trasferimenti dei dati per il carico di lavoro.

Esegui il backup dei dati solo quando sono difficili da ricreare: per ridurre al minimo l'uso delle risorse di archiviazione, esegui il backup solo dei dati che abbiano un valore aziendale o siano considerati necessari per soddisfare requisiti di conformità. Esamina le policy di backup ed escludi l'archiviazione temporanea che non offre valore in uno scenario di ripristino.

Modelli hardware

Cerca opportunità per ridurre l'impatto dei carichi di lavoro in termini di sostenibilità apportando modifiche alle tue prassi di gestione hardware. Riduci la quantità di hardware necessaria per il provisioning e l'implementazione e seleziona l'hardware più efficiente per il singolo carico di lavoro.

La seguente domanda si concentra su queste considerazioni relative alla sostenibilità:

SUS 5: In che modo la gestione dell'hardware e le procedure di utilizzo sostengono i tuoi obiettivi di sostenibilità?

Cerca opportunità per ridurre l'impatto dei carichi di lavoro in termini di sostenibilità apportando modifiche alle tue prassi di gestione hardware. Riduci la quantità di hardware necessaria per il provisioning e l'implementazione e seleziona l'hardware più efficiente per il singolo carico di lavoro.

Utilizza la quantità minima di hardware per soddisfare le tue esigenze: le funzionalità del cloud consentono di apportare modifiche frequenti alle implementazioni dei carichi di lavoro. Aggiorna i componenti distribuiti man mano che le tue esigenze cambiano.

Usa tipi di istanze con il minimo impatto: monitora costantemente il rilascio di nuovi tipi di istanza e sfrutta le migliorie in tema di efficienza energetica, inclusi i tipi di istanza progettati per supportare carichi di lavoro specifici, come la formazione del machine learning, le inferenze e la transcodifica dei video.

Utilizza servizi gestiti: i servizi gestiti consentono di affidare ad AWS la responsabilità di mantenere un utilizzo medio alto e un'ottimizzazione della sostenibilità dell'hardware implementato. Utilizza i servizi gestiti per distribuire l'impatto della sostenibilità dei servizi su tutti i tenant relativi, riducendo così il singolo contributo.

Ottimizza l'utilizzo delle GPU: le Graphics Processing Unit (GPU) possono comportare un uso energetico intensivo e molti carichi di lavoro delle GPU sono altamente variabili, come il rendering, la transcodifica e la formazione e la modellazione del machine learning. Esegui le istanze GPU solo per il tempo necessario e disattivale automaticamente quando non occorrono per ridurre la quantità di risorse utilizzate.

Modelli di sviluppo e implementazione

Cerca opportunità per ridurre l'impatto di sostenibilità apportando modifiche alle tue prassi di sviluppo, test e implementazione.

La seguente domanda si concentra su queste considerazioni relative alla sostenibilità:

SUS 6: In che modo i processi di sviluppo e implementazione adottati supportano i tuoi obiettivi di sostenibilità?

Cerca opportunità per ridurre l'impatto di sostenibilità apportando modifiche alle tue prassi di sviluppo, test e implementazione.

Adotta metodi che consentono di integrare rapidamente i miglioramenti orientati alla sostenibilità: testa e convalida potenziali modifiche di miglioramento prima di distribuirle in produzione. Tieni in considerazione il costo dei test quando calcoli il potenziale vantaggio futuro di un miglioramento. Sviluppa metodi di test a basso costo per consentire la distribuzione di piccoli miglioramenti.

Mantieni aggiornato il tuo carico di lavoro: sistemi operativi, librerie e applicazioni aggiornati possono incidere sull'efficienza dei carichi di lavoro e facilitano l'adozione di tecnologie più efficienti. Il software aggiornato potrebbe anche includere funzionalità per misurare in modo più accurato l'impatto in termini di sostenibilità del carico di lavoro, poiché i fornitori offrono caratteristiche per raggiungere i propri obiettivi di sostenibilità.

Incrementa l'utilizzo degli ambienti di sviluppo: utilizza l'automazione e l'infrastruttura come codice per rendere operativi gli ambienti di preproduzione quando necessario e dismetterli quando non vengono utilizzati. Un modello comune consiste nel pianificare periodi di disponibilità che coincidano con l'orario di lavoro dei membri del team incaricati dello sviluppo. L'ibernazione è uno strumento utile per preservare lo stato e portare rapidamente le istanze online solo quando necessario. Utilizza tipi di istanze espandibili, istanze Spot, servizi di database elastici, container e altre tecnologie per allineare la capacità di sviluppo e test all'uso.

Utilizza device farm gestite per i test: le device farm gestiste distribuiscono l'impatto di sostenibilità della produzione di hardware e dell'utilizzo delle risorse su più tenant. Le device farm gestite offrono diversi tipi di dispositivi e consentono di supportare hardware meno diffusi e di generazioni precedenti e di evitare l'impatto sulla sostenibilità dei clienti dovuti ad aggiornamenti dei dispositivi non necessari.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle best practice per la sostenibilità.

Whitepaper

- Principio della sostenibilità

Video

- The Climate Pledge

Il processo di revisione

La revisione delle architetture deve essere eseguita in modo coerente, con un approccio che non colpevolizza nessuno, ma che incoraggia ad approfondire gli argomenti. Dovrebbe essere un processo leggero (di ore, non di giorni) più simile a una conversazione che non a un audit. Lo scopo della revisione di un'architettura è identificare dei problemi critici da affrontare o aree di miglioramento. Il risultato della revisione è un insieme di azioni volte a migliorare l'esperienza di utilizzo del carico di lavoro del cliente.

Come discusso nella sezione "Architettura", ogni membro del team deve prendersi la responsabilità della qualità della sua architettura. Consigliamo che i membri del team che hanno sviluppato l'architettura usino il Canone di architettura per eseguire costantemente la revisione della loro architettura, piuttosto che fare una riunione di revisione formale. Un approccio continuo permette ai membri del team di aggiornare le risposte man mano che l'architettura evolve e migliorare l'architettura di pari passo alle funzionalità.

Il Framework AWS Well-Architected è allineato alla modalità interna di revisione dei sistemi e dei servizi di AWS. Si basa su un insieme di principi di progettazione che influenzano l'approccio architetturale e su domande che garantiscano che le persone non trascurino aree che spesso figurano nell'Analisi della causa principale (RCA). Ogni volta che si presenta un problema significativo con un sistema interno, un servizio AWS o un cliente, ci serviamo della RCA per vedere se possiamo migliorare il processo di revisione utilizzato.

Le revisioni devono essere applicate a tappe fondamentali nel ciclo di vita del prodotto, all'inizio della fase di progettazione per evitare decisioni unidirezionali che sono difficili da modificare prima della data di implementazione. (Molte decisioni sono reversibili e quindi bidirezionali. Queste decisioni possono usare un processo leggero. Le decisioni unidirezionali sono difficili o impossibile da annullare e richiedono un maggiore sforzo di indagine prima di essere adottate). Una volta entrato in produzione, il carico di lavoro continuerà ad evolversi man mano che si aggiungono nuove caratteristiche e si modificano le implementazioni tecnologiche. L'architettura del carico di lavoro cambia nel tempo. Devi seguire le best practice di igiene informatica per interrompere il degrado delle caratteristiche man mano che fai evolvere l'architettura. Man mano che l'architettura cambia, dovresti seguire un insieme di processi di igiene informatica tra cui la revisione Well-Architected.

Se vuoi utilizzare la revisione come snapshot una tantum o misura indipendente, dovrai assicurarti che alla conversazione partecipino tutte le persone appropriate. Spesso ci rendiamo conto che le revisioni sono il primo momento in cui il team comprende per davvero quello che ha implementato.

Un approccio che funziona bene per la revisione dei carichi di lavoro di un altro team consiste in una serie di conversazioni informali sull'architettura in cui ottenere le risposte alla maggior parte delle domande. Quindi puoi fare una o due riunioni di follow up in cui puoi fare chiarezza o approfondire le aree ambigue e il rischio percepito.

Ecco alcuni elementi suggeriti per le tue riunioni:

- Una sala riunioni con una lavagna
- Le stampe di tutti i grafici o delle note di progettazione
- Lista di azioni delle domande che richiedono risposte a ricerche fuori banda (ad esempio, "abbiamo abilitato la crittografia o no?")

Dopo avere completato la revisione, dovresti avere un elenco di problemi a cui assegnare delle priorità sulla base del contesto aziendale. Dovrai anche prendere in considerazione l'impatto di tali problemi sul lavoro quotidiano del tuo team. Se affronti questi problemi in anticipo puoi liberare del tempo per lavorare sulla creazione di valore aziendale anziché dedicarlo a risolvere i problemi ricorrenti. Man mano che affronti i problemi, puoi aggiornare la revisione per vedere in che modo l'architettura sta migliorando.

Il valore di una revisione è evidente dopo averne eseguita una, ma all'inizio un nuovo team potrebbe essere contrario. Ecco alcune obiezioni da gestire per istruire il team sui vantaggi di una revisione:

- "Siamo troppo occupati!" (spesso si sente questa frase quando il team si sta preparando a un grande lancio.)
  - Se ti stai preparando per un grande lancio, desidererai che tutto vada bene. La revisione ti aiuta a comprendere qualsiasi problema che potresti esserti perso.
  - Ti raccomandiamo di eseguire le revisioni all'inizio del ciclo di vita del prodotto per scoprire i rischi e sviluppare un piano di mitigazione allineato con la roadmap delle funzionalità.
- "Non abbiamo tempo per utilizzare i risultati!" (Spesso questo viene detto quando c'è un evento fisso, come il Super Bowl, di cui si sta occupando il team.)
  - Questi eventi non possono essere spostati. Vuoi davvero affrontare l'evento senza conoscere i rischi della tua architettura? Anche se non ti occupi di tutti i problemi in questione, puoi comunque disporre di playbook per affrontarli se si dovessero presentare.
- "Non vogliamo che altri scoprano i segreti della nostra implementazione di soluzioni!"
  - Se poni le domande del Framework Well-Architected, il team noterà che nessuna di esse rivela informazioni proprietarie commerciali o tecniche.

Eseguendo più revisioni con i team della tua organizzazione, potresti identificare delle aree tematiche. Ad esempio, potresti notare che un gruppo di team ha gruppi di problemi in un pilastro o un argomento specifico. Puoi gestire tutte le tue revisioni in modo olistico e identificare tutti i meccanismi, la formazione o le riunioni con gli ingegneri responsabili che possono aiutare a risolvere i problemi tematici.

Conclusione

Il Framework AWS Well-Architected best fornisce practice architetturali relative a sei pilastri per la progettazione e la gestione di sistemi affidabili, sicuri, efficienti, a costi contenuti e sostenibili nel cloud. Il canone fornisce un insieme di domande che ti permettono di eseguire la revisione di un'architettura esistente o proposta. Il canone fornisce anche un insieme di best practice AWS per ogni principio. L'utilizzo del canone nella tua architettura ti aiuta a produrre sistemi stabili ed efficienti, che ti permettono di concentrarti sui tuoi requisiti funzionali.

Collaboratori

Hanno contribuito alla stesura di questo documento:

- Brian Carlson, Operations Lead Well-Architected, Amazon Web Services
- Ben Potter, Security Lead Well-Architected, Amazon Web Services
- Seth Eliot, Reliability Lead Well-Architected, Amazon Web Services
- Eric Pullen, Sr. Solutions Architect, Amazon Web Services
- Rodney Lester, Principal Solutions Architect, Amazon Web Services
- Jon Steele, Senior Technical Account Manager, Amazon Web Services
- Max Ramsay, Principal Security Solutions Architect, Amazon Web Services
- Callum Hughes, Solutions Architect, Amazon Web Services
- Aden Leirer, Content Program Manager Well-Architected, Amazon Web Services

Approfondimenti

Centro di progettazione AWS

Conformità di AWS Cloud

Programma Partner AWS Well-Architected

AWS Well-Architected Tool

Homepage di Well-Architected AWS

Whitepaper sul principio dell'eccellenza operativa

Whitepaper sul principio della sicurezza

Whitepaper sul principio dell'affidabilità

Whitepaper sul principio dell'efficienza delle prestazioni

Whitepaper sul principio dell'ottimizzazione dei costi

Whitepaper sul principio della sostenibilità

Amazon Builders' Library

Revisioni del documento

Per ricevere una notifica sugli aggiornamenti di questo whitepaper, iscriviti al feed RSS.

Modifica	Descrizione	Data
Whitepaper aggiornato	Best practice aggiornate con nuova guida all'imple mentazione.	June 27, 2024
Aggiornamento principale	Importante principio prestazio nale modificare la struttura per portare a cinque il numero di aree di best practice. Ampio aggiornamento delle best practice e delle linee guida nel pilastro della sicurezza in Risposta agli incidenti (SEC 10). Modifiche important i ai contenuti e consolida mento nelle aree di eccellenz a operativa OPS 04, 05, 06, 08 e 09. Aggiornamenti delle linee guida relative ai pilastri di ottimizzazione dei costi e affidabilità . Aggiornamenti di minore entità ai livelli di rischio dei principi della sostenibilità .	October 3, 2023
Aggiornamenti per il nuovo canone	Best practice aggiornate con prontuario e nuove best practice aggiunte. Nuove domande aggiunte sui principi di sicurezza e di ottimizza zione dei costi.	April 10, 2023

Aggiornamento di minore entità	Aggiunta della definizione di livello di impegno e aggiornamento delle best practice nell'appendice.	October 20, 2022
Whitepaper aggiornato	Aggiunta del Principio della sostenibilità e collegamenti aggiornati.	December 2, 2021
Aggiornamento principale	Aggiunta al framework del principio della sostenibilità.	November 20, 2021
Aggiornamento di minore entità	Rimozione del linguaggio non inclusivo.	April 22, 2021
Aggiornamento di minore entità	Correzione di diversi collegamenti.	March 10, 2021
Aggiornamento di minore entità	Modifiche editoriali di minore entità in varie parti del documento.	July 15, 2020
Aggiornamenti per il nuovo canone	Revisione e riscrittura della maggior parte delle domande e delle risposte.	July 8, 2020
Whitepaper aggiornato	Aggiunta di AWS Well-Architected Tool, collegamenti ai corsi AWS Well-Architected Labs e ai partner AWS Well-Architected, correzioni minori per abilitare la versione in più lingue del canone.	July 1, 2019

Whitepaper aggiornato	Revisione e riscrittura di molte domande e risposte per garantire che le domande si concentrino su un argomento alla volta. Per questo motivo, alcune delle domande precedenti sono state divise in più domande. Aggiunta di termini comuni alle definizioni (carichi di lavoro, componenti, ecc.). Presentazione delle domande modificata per includere il testo descrittivo.	November 1, 2018
Whitepaper aggiornato	Aggiornamenti volti a semplificare il testo delle domande e a migliorare la leggibilità.	June 1, 2018
Whitepaper aggiornato	Eccellenza operativa spostata all'inizio della sezione sui pilastri e riscritta in modo che inquadri gli altri pilastri. Aggiornamenti degli altri principi per riflettere l'evoluzione di AWS.	November 1, 2017
Whitepaper aggiornato	Framework aggiornato per includere i pilastri dell'eccellenza operativa; altri pilastri rivisti e aggiornati per ridurre la duplicazione e incorporare le nozioni apprese grazie alle revisioni eseguite con migliaia di clienti.	November 1, 2016

Aggiornamenti di minore entità	Aggiornamento dell'Appendice con informazioni aggiornate su Amazon CloudWatch Logs.	November 1, 2015
Pubblicazione originale	Pubblicazione del Framework AWS Well-Architected.	October 1, 2015

Appendice: domande e best practice

Questa appendice riassume tutte le domande e le best practice nel Framework AWS Well-Architected.

Principi

- Eccellenza operativa
- Sicurezza
- Affidabilità
- Efficienza delle prestazioni
- Ottimizzazione dei costi
- Sostenibilità

Eccellenza operativa

Il principio dell'eccellenza operativa include la capacità di supportare lo sviluppo ed eseguire carichi di lavoro in modo efficace, ottenere informazioni approfondite sulle operazioni e migliorare continuamente i processi e le procedure di supporto per offrire valore commerciale. È possibile trovare linee guida prescrittive sull'implementazione nel Whitepaper sul principio dell'eccellenza operativa.

Aree delle best practice

- Organizzazione
- Preparazione
- Operatività
- Evoluzione

Organizzazione

Domande

- OPS 1. In che modo stabilisci quali sono le tue priorità?
- OPS 2. Come strutturare la tua organizzazione per supportare i risultati aziendali?
- OPS 3. In che modo la cultura aziendale supporta i risultati aziendali?

OPS 1. In che modo stabilisci quali sono le tue priorità?

È necessario che ognuno comprenda il proprio ruolo per rendere possibile il successo aziendale. Devi disporre di obiettivi comuni al fine di stabilire le priorità per le risorse. Ciò massimizzerà i risultati dei tuoi sforzi.

Best practice

- OPS01-BP01 Valutazione delle esigenze dei clienti
- OPS01-BP02 Valutazione delle esigenze dei clienti interni
- OPS01-BP03 Valutazione dei requisiti di governance
- OPS01-BP04 Valutazione dei requisiti di conformità
- OPS01-BP05 Valutazione del panorama delle minacce
- OPS01-BP06 Valutazione dei compromessi gestendo vantaggi e rischi

OPS01-BP01 Valutazione delle esigenze dei clienti

Coinvolgi le principali parti interessate, compresi i team aziendali, di sviluppo e operativi, per determinare dove concentrare gli sforzi in base alle esigenze dei clienti esterni. Avrai così una conoscenza approfondita del supporto operativo necessario per raggiungere i risultati aziendali desiderati.

Risultato desiderato:

- Lavori a ritroso partendo dalle esigenze dei clienti.
- Comprendi in che modo le procedure operative supportano i risultati e gli obiettivi aziendali.
- Coinvolgi tutte le parti interessate.
- Disponi di meccanismi per soddisfare le esigenze dei clienti.

Anti-pattern comuni:

- Hai deciso di non fornire il servizio clienti al di fuori dell'orario lavorativo di base, ma non hai esaminato i dati cronologici riguardanti le richieste di supporto. Non sai se questo determinerà un impatto sui tuoi clienti.
- Stai sviluppando una nuova funzionalità, ma non hai coinvolto i clienti per capire se è desiderata ed eventualmente in quale forma; inoltre non hai condotto attività di sperimentazione per convalidarne la necessità e il metodo di distribuzione.

Vantaggi dell'adozione di questa best practice: i clienti le cui esigenze sono soddisfatte hanno maggiori probabilità di rimanere tali. Valutando e comprendendo le esigenze dei clienti esterni sarà possibile organizzare le attività in base alle priorità e offrire valore aggiunto.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Comprendi le esigenze aziendali: il successo dell'azienda nasce dalla condivisione di obiettivi e conoscenze tra le parti interessate, compresi i team aziendali, di sviluppo e operativi.

Esamina gli obiettivi aziendali, le esigenze e le priorità dei clienti esterni: coinvolgi le principali parti interessate, compresi i team aziendali, di sviluppo e operativi, per discutere obiettivi, esigenze e priorità dei clienti esterni. Otterrai così una conoscenza approfondita del supporto operativo necessario per raggiungere i risultati aziendali e del cliente.

Condividi le conoscenze: condividi le conoscenze sulle funzioni aziendali del carico di lavoro, sui ruoli di ciascuno dei team nel gestire il carico di lavoro e su come questi supportino gli obiettivi aziendali condivisi tra clienti interni ed esterni.

Risorse

Best practice correlate:

- OPS11-BP03 Implementazione di cicli di feedback

OPS01-BP02 Valutazione delle esigenze dei clienti interni

Coinvolgi i principali stakeholder, compresi i team aziendali, di sviluppo e operativi, nel determinare dove concentrare le attività in base alle esigenze dei clienti interni. Questo ti garantirà una conoscenza approfondita del supporto operativo necessario per raggiungere i risultati aziendali.

Risultato desiderato:

- Utilizzi le priorità definite per concentrare le iniziative di miglioramento delle operazioni laddove avranno il maggiore impatto (ad esempio, sviluppare le competenze dei team, migliorare le prestazioni del carico di lavoro, ridurre i costi, automatizzare i runbook o potenziare il monitoraggio).
- Aggiorni le priorità al mutare delle esigenze.

Anti-pattern comuni:

- Per semplificare la gestione della rete hai deciso di modificare l'assegnazione degli indirizzi IP per i team di prodotto senza consultarli. Non conosci l'impatto che questo avrà sui tuoi team di prodotto.
- Stai implementando un nuovo strumento di sviluppo, ma non hai coinvolto i clienti interni per scoprire se è necessario o se è compatibile con le loro pratiche esistenti.
- Stai implementando un nuovo sistema di monitoraggio, ma non hai contattato i clienti interni per scoprire se hanno esigenze di monitoraggio o reporting da tenere in considerazione.

Vantaggi dell'adozione di questa best practice: la valutazione e la comprensione delle esigenze dei clienti interni consente di stabilire le priorità delle attività per offrire valore aziendale.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

- Comprendi le esigenze aziendali: il successo dell'azienda nasce dalla condivisione di obiettivi e conoscenze tra le parti interessate, compresi i team aziendali, di sviluppo e operativi.
- Esamina gli obiettivi aziendali, le esigenze e le priorità dei clienti interni: coinvolgi le principali parti interessate, compresi i team aziendali, di sviluppo e operativi, per discutere obiettivi, esigenze e priorità dei clienti interni. In tal modo otterrai una conoscenza approfondita del supporto operativo necessario per raggiungere i risultati aziendali e del cliente.
- Stabilisci una comprensione condivisa: stabilisci una comprensione condivisa delle funzioni aziendali del carico di lavoro, dei ruoli di ciascuno dei team nel gestire il carico di lavoro e di come questi supportino gli obiettivi aziendali condivisi tra clienti interni ed esterni.

Risorse

Best practice correlate:

- OPS11-BP03 Implementazione di cicli di feedback

OPS01-BP03 Valutazione dei requisiti di governance

Con governance si intende l'insieme di policy, regole o framework che un'azienda usa per raggiungere i propri obiettivi. I requisiti di governance vengono generati all'intero dell'organizzazione. Possono influire sui tipi di tecnologia che scegli o sul modo in cui sviluppi il tuo carico di lavoro.

Integra i requisiti di governance della tua organizzazione nel tuo carico di lavoro. La conformità è la capacità di dimostrare che hai implementato i requisiti di governance.

Risultato desiderato:

- I requisiti di governance sono integrati nel progetto architetturale e nell'operatività del tuo carico di lavoro.
- Puoi dimostrare di aver seguito i requisiti di governance.
- I requisiti di governance vengono rivisti e aggiornati con regolarità.

Anti-pattern comuni:

- La tua azienda richiede che l'account root abbia l'autenticazione multi-fattore. Non sei riuscito a implementare questo requisito e l'account root è compromesso.
- Durante la progettazione del carico di lavoro hai scelto un tipo di istanza non approvata dal dipartimento IT. Non riesci ad avviare il tuo carico di lavoro e devi procedere a una nuova progettazione.
- Devi avere un piano di ripristino di emergenza. Non ne hai uno e il tuo carico di lavoro è vittima di un'interruzione prolungata.
- Il tuo team vuole usare nuove istanze, ma i requisiti di governance non sono stati aggiornati e pertanto non sono consentite.

Vantaggi dell'adozione di questa best practice:

- Rispettare i requisiti di governance permette di allineare il carico di lavoro a policy organizzative di più ampio respiro.
- I requisiti di governance si basano su standard e best practice di settore per la tua organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

Identifica il requisito di governance collaborando con le parti interessate e le organizzazioni preposte. Includi i requisiti di governance nel tuo carico di lavoro. Dimostra di aver seguito i requisiti di governance.

Esempio del cliente

In AnyCompany Retail, il team operativo nell'ambiente cloud collabora con le parti interessate dell'organizzazione per sviluppare i requisiti di governance. Ad esempio, proibiscono l'accesso SSH alle istanze Amazon EC2. Se i team hanno necessità di accedere ai sistemi, devono usare AWS Systems Manager Session Manager. Il team operativo nell'ambiente cloud aggiorna con regolarità i requisiti di governance nel momento in cui vengono rilasciati nuovi servizi.

Passaggi dell'implementazione

1. Identifica le parti interessate per il tuo carico di lavoro, inclusi eventuali team centralizzati.
2. Collabora con le parti interessate per identificare i requisiti di governance.
3. Dopo aver generato un elenco, dai la priorità alle voci relative a migliorie e inizia a implementarle nel tuo carico di lavoro.
    a. Usa servizi come AWS Config per creare governance-as-code e verificare che tali requisiti di governance siano rispettati.
    b. Se usi AWS Organizations, puoi avvalerti di Policy di controllo dei servizi per implementare requisiti di governance.
4. Fornisci la documentazione che convalida l'implementazione.

Livello di impegno per il piano di implementazione: Medio. L'implementazione di requisiti di governance mancanti può causare la rielaborazione del tuo carico di lavoro.

Risorse

Best practice correlate:

- OPS01-BP04 Valutazione dei requisiti di conformità - La conformità è come la governance, ma è esterna all'organizzazione.

Documenti correlati:

- Guida AWS sull'ambiente cloud di governance e gestione
- Best practice per le policy di controllo dei servizi di AWS Organizations in un ambiente multi-account
- Governance nel Cloud AWS: il giusto equilibrio tra agilità e sicurezza
- Cosa si intende per Governance, Rischio e Conformità (GRC)?

Video correlati:

- Gestione e governance AWS: configurazione, conformità e revisione - AWS Online Tech Talks
- AWS re:Inforce 2019: Governance per l'età del cloud (DEM12-R1)
- AWS re:Invent 2020: Ottieni compliance as code con AWS Config
- AWS re:Invent 2020: Governance agile su AWS GovCloud (US)

Esempi correlati:

- Esempi di pacchetti di conformità AWS Config

Servizi correlati:

- AWS Config
- AWS Organizations - Policy di controllo dei servizi

OPS01-BP04 Valutazione dei requisiti di conformità

I requisiti di conformità interna, di settore e normativa sono un fattore importante per la definizione delle priorità della tua organizzazione. L'assetto di conformità della tua azienda potrebbe impedirti di usare tecnologie specifiche o posizioni geografiche. Applica la due diligence in assenza di contesti di conformità esterni. Genera revisioni o report per convalidare la conformità.

Se comunichi all'esterno che il tuo prodotto è in linea con standard specifici di conformità, devi avere un processo interno in grado di garantire costantemente che tale affermazione sia vera. Gli esempi di standard di conformità includono PCI DSS, FedRAMP e HIPAA. Gli standard di conformità applicabili vengono stabiliti in base a diversi fattori, come il tipo di dati che la soluzione archivia o trasmette e quali aree geografiche sono supportate dalla soluzione.

Risultato desiderato:

- Requisiti di conformità interni, di settore e normativi sono parte integrante della selezione dell'architettura.
- Puoi verificare la conformità e generare report di audit.

Anti-pattern comuni:

- Parti del tuo carico di lavoro rientrano nel framework Payment Card Industry Data Security Standard (PCI-DSS), ma il tuo carico di lavoro archivia dati di carte di credito non crittografati.
- Gli architetti e gli sviluppatori software non conoscono il contesto di conformità che la tua organizzazione è tenuta a rispettare.
- L'audit annuale Systems and Organizations Control (SOC2) Type II avrà luogo a breve e tu non sei in grado di verificare la presenza dei controlli richiesti.

Vantaggi dell'adozione di questa best practice:

- Valutando e comprendendo i requisiti di conformità che si applicano al carico di lavoro sarà possibile organizzare le attività in base a priorità e offrire valore aggiunto.
- Scegli le sedi e le tecnologie corrette, in linea con il tuo contesto di integrità.
- La progettazione del tuo carico di lavoro ai fini dell'auditing ti consente di dimostrare la tua aderenza al modello di conformità.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

Implementare questa best practice significa integrare requisiti di conformità nel processo di progettazione dell'architettura. I membri del tuo team sono a conoscenza del contesto di conformità richiesto. Convalidi la conformità in linea con il contesto.

Esempio del cliente

AnyCompany Retail archivia informazioni sulle carte di credito per i clienti. Gli sviluppatori del team di archiviazione delle carte sono al corrente della necessità di rispettare la conformità agli standard PCI-DSS. Hanno adottato misure per verificare che le informazioni sulle carte di credito siano archiviate e consultabili in totale sicurezza, in linea con quanto stabilito dagli standard PCI-DSS: Ogni anno collaborano con il team di sicurezza per confermare la conformità.

Passaggi dell'implementazione

1. Collabora con i team di sicurezza e governance per stabilire quali conformità interne, normative o di settore deve rispettare il tuo carico di lavoro. Integra gli standard di conformità nel tuo carico di lavoro.
   a. Convalida continuamente la conformità delle risorse AWS con servizi come AWS Compute Optimizer e AWS Security Hub.

2. Comunica ai membri del tuo team i requisiti di conformità, in modo che possano gestire e far evolvere il carico di lavoro in linea con tali requisiti. I requisiti di conformità devono essere inclusi nelle scelte tecnologiche e architetturali.
3. A seconda del contesto di conformità, potresti dover generare un report di audit o conformità. Collabora con la tua organizzazione per automatizzare il più possibile questo processo.
   a. Usa servizi come AWS Audit Manager per convalidare la conformità e generare report di audit.
   b. Puoi scaricare documenti su conformità e sicurezza AWS con AWS Artifact.

Livello di impegno per il piano di implementazione: Medio. Implementare i requisiti di conformità può essere complesso. Generare report di audit o documenti di conformità aggiunge altre complessità.

Risorse

Best practice correlate:

- SEC01-BP03 Identificazione e convalida degli obiettivi di controllo - Gli obiettivi di controllo della sicurezza sono una parte importante della conformità nel suo complesso.
- SEC01-BP06 Automatizzazione dei test e della convalida dei controlli di sicurezza nelle pipeline - Come parte delle tue pipeline, convalida i controlli di sicurezza. Puoi anche generare la documentazione di conformità per le nuove modifiche.
- SEC07-BP02 Definizione dei controlli di protezione dei dati - Molti standard di conformità si basano su policy di gestione e di archiviazione dei dati.
- SEC10-BP03 Preparazione di funzionalità forensi - Le funzionalità forensi possono a volte essere usate nella conformità di auditing.

Documenti correlati:

- AWS Compliance Center
- Risorse di conformità AWS
- Whitepaper Risk and Compliance AWS:
- Modello di responsabilità condivisa AWS
- Servizi AWS inerenti secondo i programmi di conformità

Video correlati:

- AWS re:Invent 2020: Ottieni compliance as code con AWS Compute Optimizer
- AWS re:Invent 2021 - Conformità, garanzia e auditing del cloud
- AWS Summit ATL 2022 - Implementare conformità, garanzia e auditing su AWS (COP202)

Esempi correlati:

- PCI DSS e best practice per la sicurezza di base di AWS su AWS

Servizi correlati:

- AWS Artifact
- AWS Audit Manager
- AWS Compute Optimizer
- AWS Security Hub

OPS01-BP05 Valutazione del panorama delle minacce

Valuta le minacce per l'azienda (ad esempio, concorrenza, rischi e responsabilità aziendali, rischi operativi e minacce per la sicurezza delle informazioni) e conserva le informazioni aggiornate in un registro dei rischi. Quando stabilisci dove concentrare gli sforzi, tieni in considerazione l'impatto dei rischi.

Il Framework Well-Architected favorisce l'apprendimento, la misurazione e il miglioramento. Fornisce una strategia coerente per la valutazione delle architetture e l'implementazione di progetti in grado di scalare nel corso del tempo. AWS mette a disposizione lo AWS Well-Architected Tool per aiutarti ad analizzare l'approccio prima dello sviluppo e lo stato dei carichi di lavoro prima e durante la fase di produzione. Puoi confrontare il tuo approccio con le best practice architetturali AWS più recenti, monitorare lo stato complessivo dei carichi di lavoro e ottenere approfondimenti sui potenziali rischi.

I clienti AWS possono usufruire della revisione Well-Architected dei carichi di lavoro mission-critical per misurare le loro architetture rispetto alle best practice AWS. I clienti del supporto Enterprise possono utilizzare una revisione delle operazioni ideata per agevolare l'identificazione delle lacune nell'approccio che usano nel cloud.

Il coinvolgimento trasversale dei team per tali controlli aiuta a comprendere a livello comune i carichi di lavoro e come i ruoli del team contribuiscano al successo. Le esigenze identificate nel corso dell'analisi possono aiutarti a definire le priorità.

AWS Trusted Advisor è uno strumento che fornisce l'accesso a una serie di controlli di base che propongono ottimizzazioni utili per la definizione delle priorità. I clienti del supporto Business ed Enterprise hanno accesso a ulteriori controlli a livello di sicurezza, affidabilità, prestazioni e ottimizzazione dei costi che possono essere utili per definire le priorità.

Risultato desiderato:

- Esamini e intervieni periodicamente sui risultati forniti da Well-Architected e Trusted Advisor.
- Sei a conoscenza dello stato delle patch più recenti dei servizi.
- Comprendi il rischio e l'impatto delle minacce note e intervieni di conseguenza.
- Implementi le mitigazioni necessarie.
- Comunichi azioni e contesto.

Anti-pattern comuni:

- Stai utilizzando la versione precedente di una libreria software nel tuo prodotto. Non sei a conoscenza di aggiornamenti di sicurezza alla libreria per problemi che potrebbero avere un impatto imprevisto sul carico di lavoro.
- Il tuo concorrente ha appena rilasciato una versione del proprio prodotto che risolve i reclami di molti dei tuoi clienti relativi al tuo prodotto. Non hai dato priorità alla risoluzione di questi problemi noti.
- Le autorità di regolamentazione hanno perseguito aziende come la tua che non sono conformi ai requisiti di conformità alla normativa legale. Non hai dato priorità ai requisiti di conformità in sospeso.

Vantaggi dell'adozione di questa best practice: identificando e comprendendo le minacce rivolte all'organizzazione e al carico di lavoro puoi determinare quali problematiche affrontare, la loro priorità e le risorse necessarie per farlo.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Valuta il panorama delle minacce: valuta le minacce per l'azienda, ad esempio concorrenza, rischi e responsabilità aziendali, rischi operativi e minacce alla sicurezza delle informazioni, in modo da poterne includere l'impatto quando determini dove concentrare le attività.
  - Ultimi bollettini sulla sicurezza AWS

    - AWS Trusted Advisor
- Mantieni un modello delle minacce: definisci e mantieni un modello delle minacce che identifichi le potenziali minacce, le mitigazioni pianificate e predisposte nonché la loro priorità. Esamina la probabilità che le minacce si manifestino come incidenti, il costo del recupero dagli incidenti, il danno previsto causato e il costo per prevenire tali incidenti. Modifica le priorità man mano che i contenuti del modello di minaccia cambiano.

Risorse

Best practice correlate:

- SEC01-BP07 Identificare le minacce e dare priorità alle mitigazioni utilizzando un modello di minaccia

Documenti correlati:

- Conformità di Cloud AWS
- Ultimi bollettini sulla sicurezza AWS
- AWS Trusted Advisor

Video correlati:

- AWS re:Inforce 2023 - A tool to help improve your threat modeling

OPS01-BP06 Valutazione dei compromessi gestendo vantaggi e rischi

Gli interessi contrastanti di più parti possono rendere difficile l'assegnazione delle priorità agli impegni, lo sviluppo delle capacità e il conseguimento di risultati in linea con le strategie aziendali. Ad esempio, è possibile che ti venga chiesto di accelerare la commercializzazione di nuove funzionalità piuttosto che ottimizzare i costi dell'infrastruttura IT. Questa richiesta può mettere due parti interessate in conflitto tra loro. In queste situazioni, le decisioni devono essere prese da un'autorità superiore che risolve il conflitto. I dati sono necessari per rimuovere l'attaccamento emotivo dal processo decisionale.

La stessa sfida può verificarsi a livello strategico. Ad esempio, la scelta tra l'utilizzo di tecnologie di database relazionali o non relazionali può avere un impatto significativo sul funzionamento di un'applicazione. È fondamentale comprendere i risultati prevedibili delle varie scelte.

AWS può aiutarti a istruire i team su AWS e i suoi servizi, affinché comprendano meglio in che modo le loro scelte possono influire sul carico di lavoro. Per istruire i team, utilizza le risorse fornite da AWS Support (Centro conoscenze AWS, AWS Discussion Forums e AWS Support Center) e la documentazione AWS. Per ulteriori domande, contatta AWS Support.

AWS condivide inoltre best practice e modelli operativi nella Amazon Builders' Library. Un'ampia gamma di altre informazioni utili è disponibile tramite il blog AWS e il podcast ufficiale di AWS.

Risultato desiderato: disponi di un framework di governance del processo decisionale chiaramente definito per facilitare le decisioni importanti a tutti i livelli dell'organizzazione di distribuzione del cloud. Questo framework include funzionalità come un registro dei rischi, i ruoli definiti autorizzati a prendere decisioni e i modelli prestabiliti per ogni livello di decisione che può essere presa. Il framework definisce in anticipo come vengono risolti i conflitti, quali dati devono essere presentati e come viene stabilita la priorità delle opzioni, in modo che una volta prese le decisioni sia subito possibile lavorare per applicarle. Il framework del processo decisionale include un approccio standardizzato alla revisione e alla valutazione dei vantaggi e dei rischi di ogni decisione per comprenderne i compromessi. Possono essere inclusi fattori esterni, come l'aderenza ai requisiti di conformità normativa.

Anti-pattern comuni:

- Gli investitori richiedono di dimostrare la conformità agli standard PCI DSS (Payment Card Industry Data Security Standard). Non prendi in considerazione i compromessi tra soddisfare la loro richiesta e continuare con le attività di sviluppo già in corso. Al contrario, prosegui con il lavoro di sviluppo senza dimostrare la conformità. Gli investitori interrompono il supporto all'azienda per i dubbi relativi alla sicurezza della piattaforma e dei loro investimenti.
- Hai deciso di includere una libreria che uno dei tuoi sviluppatori ha trovato su Internet. Non hai valutato i rischi derivanti dall'adozione di questa libreria da un'origine sconosciuta e non sai se contiene vulnerabilità o codice dannoso.
- La giustificazione aziendale originale per la migrazione si basava sulla modernizzazione del 60% dei carichi di lavoro delle applicazioni. Tuttavia, a causa di difficoltà tecniche, è stata presa la decisione di modernizzare solo il 20%, con una riduzione dei vantaggi pianificati a lungo termine, un maggiore impegno operativo dei team dell'infrastruttura per supportare manualmente i sistemi legacy e un accresciuto affidamento sullo sviluppo di nuove competenze nei team dell'infrastruttura che non avevano pianificato questo cambiamento.

Vantaggi dell'adozione di questa best practice: ottieni un totale allineamento e il supporto per le priorità aziendali a livello dirigenziale, comprendi i rischi per raggiungere il risultato, prendi decisioni informate e agisci in modo appropriato quando i rischi ostacolano le opportunità di successo. Comprendere le implicazioni e le conseguenze delle tue decisioni ti aiuta a stabilire le priorità delle opzioni e a ottenere l'accordo dei leader più rapidamente, fornendo risultati aziendali migliori. Identifichi i vantaggi derivanti dalle tue scelte e riconosci i rischi per l'organizzazione in modo da prendere decisioni basate sui dati, piuttosto che basate su aneddoti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La gestione dei vantaggi e dei rischi deve essere definita da un organo direttivo che stabilisca i requisiti del processo decisionale chiave. Le decisioni devono essere prese e assegnate alle priorità in base ai vantaggi che ne derivano per l'organizzazione, con una comprensione dei rischi connessi. L'accuratezza delle informazioni è fondamentale quando si prendono le decisioni organizzative. Deve basarsi su misurazioni solide ed essere definita secondo le procedure comuni del settore per l'analisi costi-benefici. Per prendere questo tipo di decisioni, occorre trovare un equilibrio tra l'autorità centralizzata e quella decentralizzata. C'è sempre un compromesso ed è importante capire in che modo ogni scelta influisce sulle strategie definite e sui risultati aziendali desiderati.

Passaggi dell'implementazione

1. Formalizza le procedure di misurazione dei vantaggi in un framework olistico di governance del cloud.
   a. Bilancia il controllo centrale del processo decisionale con l'autorità decentralizzata per alcune decisioni.
   b. Comprendi che i gravosi processi decisionali imposti per ogni decisione possono rallentare le operazioni.
   c. Incorpora nel processo decisionale i fattori esterni, come i requisiti di conformità.
2. Stabilisci un framework del processo decisionale concordato per vari livelli di decisioni, che includa chi è tenuto a prendere le decisioni soggette a conflitti di interessi.
   a. Centralizza le decisioni definitive che potrebbero essere irreversibili.
   b. Consenti ai leader dell'organizzazione di livello inferiore di prendere decisioni reversibili.
3. Comprendi e gestisci i vantaggi e i rischi. Bilancia i vantaggi delle decisioni rispetto ai rischi connessi.

a. Identifica i vantaggi: identifica i vantaggi in base agli obiettivi aziendali, alle esigenze e alle priorità, ad esempio l'impatto del caso aziendale, il time-to-market, la sicurezza, l'affidabilità, le prestazioni e i costi.

b. Identifica i rischi: identifica i rischi in base agli obiettivi aziendali, alle esigenze e alle priorità, ad esempio il time-to-market, la sicurezza, l'affidabilità, le prestazioni e il costo.

c. Valuta i vantaggi rispetto ai rischi e prendi decisioni informate: determina l'impatto dei vantaggi e dei rischi in base agli obiettivi, alle esigenze e alle priorità delle principali parti interessate, inclusi business, sviluppo e operazioni. Valuta il valore del vantaggio rispetto alla probabilità di realizzazione del rischio e al costo del suo impatto. Ad esempio, enfatizzare la velocità di accesso al mercato rispetto all'affidabilità potrebbe offrire un vantaggio competitivo. Tuttavia, potrebbe causare tempi di attività ridotti in presenza di problemi di affidabilità.

4. Applica in modo programmatico le decisioni chiave che automatizzano l'aderenza ai requisiti di conformità.
5. Impiega le funzionalità e i framework noti del settore, come Value Stream Analysis e LEAN, per definire la linea di base per le prestazioni dello stato attuale, le metriche aziendali e le iterazioni dei progressi verso il miglioramento di queste metriche.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS01-BP05 Valutazione del panorama delle minacce

Documenti correlati:

- Elementi della cultura del Giorno 1 di Amazon | Prendi decisioni rapide e di qualità
- Governance del cloud
- Management and Governance Cloud Environment
- Governance in the Cloud and in the Digital Age: Parts One and Two

Video correlati:

- Podcast | Jeff Bezos | On how to make decisions

Esempi correlati:

- Make informed decisions using data (The DevOps Sagas)
- Using development value stream mapping to identify constraints to DevOps outcomes

OPS 2. Come strutturare la tua organizzazione per supportare i risultati aziendali?

I tuoi team devono comprendere quale contributo offrono nel raggiungimento dei risultati aziendali. I team devono avere obiettivi condivisi e comprendere il proprio ruolo nel successo degli altri team. Comprendere la responsabilità, la proprietà, il modo in cui vengono prese le decisioni e chi ha l'autorità decisionale aiuterà a concentrare gli sforzi e a ottimizzare i contributi dei team.

Best practice

- OPS02-BP01 Associazione di proprietari identificati alle risorse
- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure
- OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni
- OPS02-BP04 Definizione di meccanismi per gestire responsabilità e titolarità
- OPS02-BP05 Definizione di meccanismi per richiedere aggiunte, modifiche ed eccezioni
- OPS02-BP06 Predefinizione o negoziazione delle responsabilità tra i team

OPS02-BP01 Associazione di proprietari identificati alle risorse

Le risorse per il tuo carico di lavoro devono disporre di proprietari identificati per il controllo delle modifiche, la risoluzione dei problemi e altre funzioni. I titolari sono assegnati a carichi di lavoro, account, infrastrutture, piattaforme e applicazioni. La proprietà viene registrata tramite strumenti come un registro centrale o metadati collegati alle risorse. Il valore aziendale dei componenti è alla base dei processi e delle procedure applicate.

Risultato desiderato:

- Le risorse hanno identificato i titolari tramite i metadati o un registro centrale.
- I membri del team possono identificare chi è il titolare delle risorse.
- Gli account hanno un solo proprietario, laddove possibile.

Anti-pattern comuni:

- I contatti alternativi per il tuo Account AWS non sono inseriti.
- Le risorse non hanno tag che identificano i team che le possiedono.
- Hai una coda ITSM senza una mappatura delle e-mail.
- Due team hanno entrambi la proprietà di una parte critica dell'infrastruttura.

Vantaggi dell'adozione di questa best practice:

- Il controllo delle modifiche per le risorse è immediato con la proprietà assegnata.
- Puoi coinvolgere i proprietari corretti quando risolvi i problemi.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

Definisci qual è il significato della proprietà per i casi d'uso delle risorse nel tuo ambiente. Proprietà significa supervisionare le modifiche alla risorsa, supportare la risorsa durante la risoluzione dei problemi o essere finanziariamente affidabile. Specifica e registra i proprietari delle risorse, con nome, informazioni di contatto, organizzazione e team.

Esempio del cliente

AnyCompany Retail definisce il proprietario come il team o l'individuo responsabile delle modifiche e del supporto per le risorse. Si avvale di AWS Organizations per gestire gli Account AWS. Contatti alternativi degli account vengono configurati con caselle di posta di gruppo. Ogni coda ITSM è mappata su un alias e-mail. I tag identificano il proprietario delle risorse AWS. Per altre piattaforme e infrastrutture hanno una pagina wiki che identifica la proprietà e le informazioni di contatto.

Passaggi dell'implementazione

1. Inizia definendo la proprietà dell'organizzazione. La proprietà può significare essere proprietari del rischio collegato alla risorsa, essere proprietari delle modifiche alla risorsa o supportare la risorsa durante la risoluzione dei problemi. Proprietà può anche significare proprietà amministrativa o finanziaria della risorsa.
2. Usa AWS Organizations per gestire gli account. Puoi gestire centralmente i contatti alternativi per gli account.
   a. Se usi indirizzi e-mail e numeri di telefono aziendali come informazioni di contatto, puoi accedervi anche se le persone a cui appartengono non fanno più parte dell'organizzazione. Ad esempio, crea elenchi di distribuzione delle e-mail separati per fatturazione, operazioni e

sicurezza e configurali come contatti per Fatturazione, Sicurezza e Operazioni in ogni Account AWS attivo. Molteplici persone riceveranno notifiche AWS e potranno rispondere, anche se qualcuno è in vacanza, ha cambiato ruolo o ha lasciato l'azienda.

b. Se un account non è gestito da AWS Organizations, i contatti alternativi dell'account consentono ad AWS di comunicare con il personale richiesto, se necessario. Configura i contatti alternativi dell'account per indirizzare le persone a un gruppo invece che a un individuo.

3. Usa i tag per identificare i proprietari delle risorse AWS. Puoi specificare sia i proprietari sia le loro informazioni di contatto in tag separati.
   a. Puoi usare le regole AWS Config per avere la certezza che le risorse abbiano i tag di proprietà richiesti.
   b. Per linee guida dettagliate su come creare una strategia per l'applicazione di tag per l'organizzazione, consulta il whitepaper Best Practices for Tagging AWS Resources.
4. Usa Amazon Q Business, un assistente conversazionale che utilizza l'IA generativa per migliorare la produttività della forza lavoro, rispondere alle domande e completare le attività in base alle informazioni presenti nei sistemi aziendali.
   a. Connetti Amazon Q Business all'origine dati dell'azienda. Amazon Q Business offre connettori predefiniti per oltre 40 origini dati supportate, tra cui Amazon Simple Storage Service (Amazon S3), Microsoft SharePoint, Salesforce e Atlassian Confluence. Per ulteriori informazioni, consulta Connettori di Amazon Q Business.
5. Per altre risorse, piattaforme e infrastrutture, crea la documentazione che stabilisce la proprietà. Tutti i membri del team devono poter accedere a queste informazioni.

Livello di impegno per il piano di implementazione: Basso. Sfrutta le informazioni di contatto e i tag degli account per assegnare la proprietà delle risorse AWS. Per altre risorse puoi usare qualcosa di semplice come una tabella in un wiki per registrare le informazioni su proprietà e contatti o usare uno strumento ITSM per mappare la proprietà.

Risorse

Best practice correlate:

- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure
- OPS02-BP04 Definizione di meccanismi per gestire responsabilità e proprietà

Documenti correlati:

- AWS Account Management - Updating contact information
- AWS Organizations - Updating alternative contacts in your organization
- Whitepaper Best Practices for Tagging AWS Resources
- Build private and secure enterprise generative AI apps with Amazon Q Business and AWS IAM Identity Center
- Amazon Q Business, now generally available, helps boost workforce productivity with generative AI
- Cloud AWS Operations & Migrations Blog - Implementing automated and centralized tagging controls with AWS Config and AWS Organizations
- AWS Security Blog - Extend your pre-commit hooks with AWS CloudFormation Guard
- AWS DevOps Blog - Integrating AWS CloudFormation Guard into CI/CD pipelines

Workshop correlati:

- AWS Workshop - Tagging

Esempi correlati:

- Regole di AWS Config - Amazon EC2 with required tags and valid values

Servizi correlati:

- Regole di AWS Config - required-tags
- AWS Organizations

OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure

È utile comprendere chi ha la proprietà della definizione di singoli processi e procedure, perché tali processi e procedure specifici vengono utilizzati e perché tale proprietà esiste. Comprendere i motivi per cui vengono utilizzati processi e procedure specifici aiuta a identificare le opportunità di miglioramento.

Risultato desiderato: l'organizzazione dispone di una serie di processi e procedure per le attività operative ben definiti e gestiti. I processi e le procedure sono archiviati in una posizione centrale e disponibili per i membri del team. I processi e le procedure vengono aggiornati frequentemente

attraverso l'assegnazione chiara della proprietà. Ove possibile, script, modelli e documenti di automazione vengono implementati come codice.

Anti-pattern comuni:

- I processi non sono documentati. Possono essere presenti script frammentati su workstation degli operatori isolate.
- La conoscenza relativa all'uso degli script è nelle mani di pochi individui oppure viene acquisita in modo informale come conoscenza di team.
- È necessario aggiornare un processo legacy, ma la proprietà dell'aggiornamento non è chiara e l'autore originale non fa più parte dell'organizzazione.
- I processi e gli script non sono individuabili, quindi non sono immediatamente disponibili quando necessario (ad esempio, in risposta a un incidente).

Vantaggi dell'adozione di questa best practice:

- I processi e le procedure incentivano l'impegno nella gestione dei carichi di lavoro.
- I nuovi membri del team diventano efficienti più rapidamente.
- Tempi ridotti per mitigare gli incidenti.
- Membri del team (e team) diversi possono utilizzare gli stessi processi e procedure in modo coerente.
- I team scalano i processi tramite processi ripetibili.
- Processi e procedure standardizzati aiutano a mitigare l'impatto del trasferimento delle responsabilità del carico di lavoro tra i team.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

- Esistono proprietari identificati di processi e procedure, che sono responsabili della loro definizione.
  - Identifica le attività operative eseguite a supporto dei carichi di lavoro. Documenta queste attività in un percorso individuabile.
  - Identifica in modo univoco la persona o il team responsabile della specifica di un'attività. Questo soggetto deve verificare che essa possa essere eseguita correttamente dal componente di un team adeguatamente qualificato, che disponga di autorizzazioni, accesso e strumenti adeguati.

In caso di problemi nello svolgimento di tale attività, i membri del team che la eseguono sono responsabili della redazione dei feedback dettagliati necessari per migliorarla.

- Acquisisci la responsabilità dei metadati dell'artefatto dell'attività tramite servizi come AWS Systems Manager, documenti e AWS Lambda. Acquisisci la responsabilità delle risorse utilizzando tag o gruppi di risorse, specificando proprietà e informazioni di contatto. Utilizza AWS Organizations per creare policy di tagging e garantire l'acquisizione di proprietà e informazioni di contatto.

- Nel tempo, queste procedure si evolvono per essere eseguibili come codice, riducendo la necessità dell'intervento umano.
  - Ad esempio, prendi in considerazione le funzioni AWS Lambda, i modelli CloudFormation o i documenti di automazione AWS Systems Manager.
  - Esegui il controllo delle versioni nei repository appropriati.
  - Applica i tag adeguati alle risorse, in modo da facilitare l'identificazione di proprietari e documentazione.

Esempio del cliente

AnyCompany Retail definisce come titolare il team o l'individuo responsabile dei processi per un'applicazione o gruppi di applicazioni (che condividono procedure e tecnologie architetturali comuni). Inizialmente, i processi e le procedure sono documentati nel sistema di gestione dei documenti come guide dettagliate, individuabili tramite i tag dell'Account AWS che ospita l'applicazione e di gruppi specifici di risorse all'interno dell'account. Si avvalgono di AWS Organizations per gestire gli Account AWS. Nel tempo, questi processi vengono convertiti in codice e le risorse vengono definite utilizzando l'infrastructure as code (ad esempio CloudFormation o modelli AWS Cloud Development Kit (AWS CDK)). I processi operativi diventano documenti di automazione in AWS Systems Manager o nelle funzioni AWS Lambda, che possono essere avviati come attività pianificate in risposta a eventi, ad esempio allarmi AWS CloudWatch o eventi AWS EventBridge, oppure attivati da richieste di una piattaforma di gestione dei servizi IT (ITSM). Tutti i processi dispongono dei tag per identificare la titolarità. La documentazione per l'automazione e il processo viene mantenuta all'interno delle pagine wiki generate dal repository di codice per il processo.

Passaggi dell'implementazione

1. Documenta i processi e le procedure esistenti.
   a. Rivedili e mantienili aggiornati.
   b. Identifica un proprietario per ogni processo o procedura.

c. Applica a ognuno il controllo della versione.

d. Ove possibile, condividi processi e procedure tra carichi di lavoro e ambienti che condividono progetti architetturali.

2. Stabilisci meccanismi di feedback e miglioramento.

a. Definisci policy relative alla frequenza con cui i processi devono essere rivisti.

b. Definisci i processi per revisori e approvatori.

c. Implementa i problemi o crea una coda di ticket per fornire e monitorare il feedback.

d. Ove possibile, i processi e le procedure devono essere approvati preventivamente e classificati in base ai rischi da parte di un comitato di approvazione delle modifiche (CAB).

3. Verifica che i processi e le procedure siano accessibili e individuabili da chi deve eseguirli.

a. Utilizza i tag per indicare dove è possibile accedere ai processi e alle procedure per il carico di lavoro.

b. Utilizza messaggi di errore ed eventi significativi per indicare i processi o le procedure appropriati per risolvere un problema.

c. Usa i wiki e la gestione dei documenti per rendere i processi e le procedure consultabili in modo coerente in tutta l'organizzazione.

4. Automatizza quando appropriato.

a. È opportuno eseguire le automazioni quando servizi e tecnologie forniscono un'API.

b. Istruisci adeguatamente in merito ai processi. Sviluppa i casi utente e i requisiti per automatizzare i processi.

c. Misura correttamente l'uso di processi e procedure e sfrutta i problemi come un'opportunità di miglioramento continuo.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS02-BP01 Associazione di titolari identificati alle risorse
- OPS02-BP04 Definizione di meccanismi per gestire responsabilità e titolarità
- OPS11-BP04 Gestione delle conoscenze

Documenti correlati:

- Whitepaper AWS - Introduction to DevOps on AWS
- AWS Whitepaper - Best Practices for Tagging AWS Resources
- AWS Whitepaper - Organizing Your AWS Environment Using Multiple Accounts
- Cloud AWS Operations & Migrations Blog - Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services
- Cloud AWS Operations & Migrations Blog - Implementing automated and centralized tagging controls with AWS Config and AWS Organizations
- AWS Security Blog - Extend your pre-commit hooks with AWS CloudFormation Guard
- AWS DevOps Blog - Integrating AWS CloudFormation Guard into CI/CD pipelines

Workshop correlati:

- AWS Well-Architected Operational Excellence Workshop
- AWS Workshop - Tagging

Video correlati:

- How to automate IT Operations on AWS
- AWS re:Invent 2020 - Automate anything with AWS Systems Manager
- AWS re:Inforce 2022 - Automating patch management and compliance using AWS (NIS306)
- AWS Supports You - Diving Deep into AWS Systems Manager

Servizi correlati:

- AWS Systems Manager - Automation
- AWS Service Management Connector

OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni

È utile comprendere chi ha la responsabilità di eseguire attività specifiche su carichi di lavoro definiti e perché tale responsabilità esiste. Comprendere chi ha la responsabilità di eseguire le attività fornisce indicazioni su chi eseguirà l'attività, su chi convaliderà il risultato e su chi fornirà feedback al titolare dell'attività.

Risultato desiderato:

L'organizzazione definisce chiaramente le responsabilità per eseguire attività specifiche su carichi di lavoro stabiliti e rispondere agli eventi generati dai carichi di lavoro. L'organizzazione documenta la responsabilità dei processi e degli impegni e rende queste informazioni individuabili. Esamini e aggiorni le responsabilità quando avvengono cambiamenti organizzativi e i team monitorano e misurano le prestazioni delle attività di identificazione dei difetti e delle inefficienze. Implementi i meccanismi di feedback per monitorare i difetti e i miglioramenti e supportare il miglioramento continuo.

Anti-pattern comuni:

- Non documenti le responsabilità.
- Sono presenti script frammentati sulle workstation degli operatori isolate. Solo poche persone sanno come usarli o li chiamano informalmente conoscenze del team.
- Un processo legacy deve essere aggiornato, ma non si sa chi è il proprietario e l'autore originale non fa più parte dell'organizzazione.
- I processi e gli script non sono individuabili, quindi non sono immediatamente disponibili quando necessario, ad esempio, in risposta a un incidente.

Vantaggi dell'adozione di questa best practice:

- Comprendi chi è responsabile dell'esecuzione di un'attività, a chi notificare un'azione necessaria e chi esegue l'azione, convalida il risultato e fornisce il feedback al titolare dell'attività.
- I processi e le procedure incentivano l'impegno nella gestione dei carichi di lavoro.
- I nuovi membri del team diventano efficienti più rapidamente.
- Riduci il tempo necessario per mitigare gli incidenti.
- Team diversi utilizzano medesimi processi e procedure per eseguire le attività in modo coerente.
- I team scalano i processi tramite processi ripetibili.
- Processi e procedure standardizzati aiutano a mitigare l'impatto del trasferimento delle responsabilità del carico di lavoro tra i team.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per definire le responsabilità, inizia usando la documentazione esistente, ad esempio matrici di responsabilità, processi e procedure, ruoli e responsabilità, strumenti e automazione. Esamina la documentazione e organizza discussioni sulle responsabilità dei processi documentati. Collaborando con i team identifica i disallineamenti tra le responsabilità e i processi documentati. Parla dei servizi offerti con i clienti interni di un team per identificare le divergenze nelle aspettative tra i team.

Analizza e risolvi le discrepanze. Identifica le opportunità di miglioramento e individua le attività richieste di frequente e con uso intensivo di risorse, che in genere sono ottime candidate al miglioramento. Esamina le best practice, i modelli e le linee guida prescrittive per semplificare e standardizzare i miglioramenti. Registra le opportunità di miglioramento e monitora i miglioramenti fino al completamento.

Nel tempo, queste procedure si evolvono per essere eseguibili come codice, riducendo la necessità dell'intervento umano. Ad esempio, le procedure possono essere avviate come funzioni AWS Lambda, modelli AWS CloudFormation o documenti di automazione AWS Systems Manager. Verifica che queste procedure siano sottoposte al controllo delle versioni nei repository appropriati e includano i corretti tag delle risorse in modo che i team possano identificare prontamente i responsabili e la documentazione. Documenta la responsabilità dello svolgimento delle attività, quindi monitora l'avvio e il funzionamento delle automazioni, nonché le prestazioni dei risultati desiderati.

Esempio del cliente

AnyCompany Retail definisce come titolare il team o l'individuo responsabile dei processi per un'applicazione o gruppi di applicazioni (che condividono procedure e tecnologie architetturali comuni). Inizialmente, l'azienda documenta i processi e le procedure come guide dettagliate nel sistema di gestione dei documenti. Rende le procedure individuabili applicando i tag nell'Account AWS che ospita l'applicazione e in gruppi specifici di risorse dell'account, utilizzando AWS Organizations per gestire gli Account AWS. Nel tempo, AnyCompany Retail converte questi processi in codice e definisce le risorse utilizzando l'infrastructure as code, tramite servizi come CloudFormation o modelli AWS Cloud Development Kit (AWS CDK). I processi operativi diventano documenti di automazione in AWS Systems Manager o nelle funzioni AWS Lambda, che possono essere avviati come attività pianificate in risposta a eventi, ad esempio allarmi Amazon CloudWatch o eventi Amazon EventBridge, oppure da richieste di una piattaforma di gestione dei servizi IT (ITSM). Tutti i processi dispongono dei tag per identificare il proprietario. I team gestiscono la documentazione per l'automazione e il processo nelle pagine wiki generate dal repository di codice per il processo.

Passaggi dell'implementazione

1. Documenta i processi e le procedure esistenti.
   a. Esamina e verifica che siano aggiornati.
   b. Verifica che ogni processo o procedura abbia un titolare.
   c. Applica alle procedure il controllo delle versioni.
   d. Ove possibile, condividi processi e procedure tra carichi di lavoro e ambienti che condividono strutture architetturali.
2. Stabilisci meccanismi di feedback e miglioramento.
   a. Definisci policy relative alla frequenza con cui i processi devono essere rivisti.
   b. Definisci i processi per revisori e approvatori.
   c. Implementa i problemi o crea una coda di ticket per fornire e monitorare il feedback.
   d. Ove possibile, i processi e le procedure devono essere approvati preventivamente e classificati in base ai rischi da parte di un comitato di approvazione delle modifiche (CAB).
3. Rendi i processi e le procedure accessibili e individuabili dagli utenti che devono eseguirli.
   a. Utilizza i tag per indicare dove è possibile accedere ai processi e alle procedure per il carico di lavoro.
   b. Utilizza messaggi di errore ed eventi significativi per indicare i processi o le procedure appropriati per risolvere il problema.
   c. Usa i wiki o la gestione dei documenti per rendere i processi e le procedure consultabili in modo coerente in tutta l'organizzazione.
4. Automatizza quando è opportuno farlo.
   a. Laddove servizi e tecnologie forniscono un'API, sviluppa le automazioni.
   b. Verifica che i processi siano ben compresi e sviluppa i casi utente e i requisiti per automatizzare i processi.
   c. Misura l'uso corretto di processi e procedure e sfrutta i problemi per supportare il miglioramento continuo.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS02-BP01 Associazione di titolari identificati alle risorse
- OPS02-BP02 Assegnazione di titolari identificati a processi e procedure
- OPS02-BP04 Definizione di meccanismi per gestire responsabilità e titolarità
- OPS02-BP05 Definizione di meccanismi per identificare responsabilità e titolarità
- OPS11-BP04 Gestione delle conoscenze

Documenti correlati:

- Whitepaper AWS | Introduction to DevOps on AWS
- AWS Whitepaper | Best Practices for Tagging AWS Resources
- AWS Whitepaper | Organizing Your AWS Environment Using Multiple Accounts
- Cloud AWS Operations & Migrations Blog | Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services
- AWS Workshop - Tagging
- AWS Service Management Connector

Video correlati:

- AWS Knowledge Center Live | Tagging AWS Resources
- AWS re:Invent 2020 | Automate anything with AWS Systems Manager
- AWS re:Inforce 2022 | Automating patch management and compliance using AWS (NIS306)
- AWS Supports You | Diving Deep into AWS Systems Manager

Esempi correlati:

- AWS Well-Architected Operational Excellence Workshop

OPS02-BP04 Definizione di meccanismi per gestire responsabilità e titolarità

Comprendi le responsabilità del tuo ruolo e il modo in cui contribuisci ai risultati aziendali in quanto questa conoscenza fornisce indicazioni sulle priorità delle tue attività e sul perché il tuo ruolo è importante. I membri del team possono quindi riconoscere le esigenze e rispondere in modo appropriato. Quando i membri del team comprendono il proprio ruolo, possono stabilire la titolarità,

identificare le opportunità di miglioramento e capire come influenzare o apportare le modifiche appropriate.

Occasionalmente, una responsabilità potrebbe non avere un titolare definito. In queste situazioni, progetta un meccanismo per risolvere la lacuna. Crea un percorso di escalation ben definito a qualcuno con l'autorità di assegnare la responsabilità o il piano per risolvere il problema.

Risultato desiderato: i team all'interno dell'organizzazione hanno responsabilità chiaramente definite che includono il modo in cui sono correlate alle risorse, alle azioni da eseguire, ai processi e alle procedure. Queste responsabilità sono in linea con le responsabilità e gli obiettivi del team, nonché con le responsabilità degli altri team. Documenti i percorsi di escalation in modo coerente e individuabile e inserisci queste decisioni in artefatti di documentazione, come matrici di responsabilità, definizioni di team o pagine wiki.

Anti-pattern comuni:

- Le responsabilità del team sono ambigue o mal definite.
- Il team non allinea i ruoli alle responsabilità.
- Il team non allinea scopi e obiettivi alle responsabilità, rendendo difficile misurare il successo delle attività.
- Le responsabilità dei membri del team non sono in linea con il team e l'organizzazione in generale.
- Il team non mantiene aggiornate le responsabilità rendendole incoerenti con le attività svolte dal team.
- I percorsi di escalation per determinare le responsabilità non sono definiti o non sono chiari.
- I percorsi di escalation non hanno un unico responsabile del thread per garantire una risposta tempestiva.
- Ruoli, responsabilità e percorsi di escalation non sono individuabili e quindi non sono immediatamente disponibili quando richiesto, ad esempio in risposta a un incidente.

Vantaggi dell'adozione di questa best practice:

- Una volta compreso chi ha la responsabilità o la titolarità, puoi contattare il team o il membro del team appropriato per effettuare una richiesta o trasferire un'attività.
- Per ridurre il rischio di inattività e di esigenze non soddisfatte, identifichi una persona che ha l'autorità di assegnare responsabilità o titolarità.

- Quando si definisce chiaramente l'ambito di una responsabilità, i membri del team acquisiscono autonomia e titolarità.
- Le tue responsabilità forniscono indicazioni sulle decisioni che prendi, sulle azioni che intraprendi e sulle tue attività di distribuzione ai titolari appropriati.
- Ti sarà facile identificare le responsabilità abbandonate perché hai una chiara comprensione di ciò che non rientra nelle responsabilità del tuo team e quindi potrai effettuare l'escalation per chiedere chiarimenti.
- I team evitano confusione e tensione e possono gestire in modo più adeguato i carichi di lavoro e le risorse.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Identifica i ruoli e le responsabilità dei membri del team e verifica che comprendano le aspettative del proprio ruolo. Rendi queste informazioni individuabili in modo che i membri della tua organizzazione possano identificare il team o la persona da contattare per esigenze specifiche. Man mano che le organizzazioni capitalizzano le opportunità di migrazione e modernizzazione su AWS, i ruoli e le responsabilità potrebbero cambiare. Rendi i team e i membri consapevoli delle loro responsabilità e offri la formazione appropriata per svolgere le attività durante questo cambiamento.

Determina il ruolo o il team che deve ricevere le escalation per identificare responsabilità e titolarità. Questo team può interagire con varie parti interessate per prendere le decisioni. Tuttavia, è proprietario della gestione del processo decisionale.

Fornisci ai membri della tua organizzazione meccanismi accessibili per scoprire e identificare titolarità e responsabilità. Questi meccanismi insegnano loro a chi rivolgersi per esigenze specifiche.

Esempio del cliente

AnyCompany Retail ha recentemente completato una migrazione dei carichi di lavoro da un ambiente on-premises alla zona di destinazione in AWS con un approccio lift and shift. Ha eseguito una revisione delle operazioni per esaminare come vengono svolte le attività operative comuni e ha verificato che la matrice di responsabilità esistente rifletta le operazioni nel nuovo ambiente. Quando ha eseguito la migrazione dall'ambiente on-premises ad AWS, ha ridotto le responsabilità dei team dell'infrastruttura relative all'hardware e all'infrastruttura fisica. Questo passaggio ha anche rivelato nuove opportunità per evolvere il modello operativo dei carichi di lavoro.

Oltre ad aver identificato, risolto e documentato la maggior parte delle responsabilità, ha anche definito i percorsi di escalation per eventuali responsabilità mancanti o che potrebbero cambiare con l'evolversi delle procedure operative. Per la ricerca di nuove opportunità per standardizzare e migliorare l'efficienza dei carichi di lavoro, fornisce l'accesso a strumenti operativi come AWS Systems Manager e strumenti di sicurezza come AWS Security Hub e Amazon GuardDuty. AnyCompany Retail combina una revisione delle responsabilità e della strategia sulla base dei miglioramenti che intende eseguire per primi. Man mano che l'azienda adotta nuovi modi di lavorare e modelli tecnologici, aggiorna la propria matrice di responsabilità di conseguenza.

Passaggi dell'implementazione

1. Inizia con la documentazione esistente. Alcuni documenti di origine tipici possono essere:
    a. Matrici di responsabilità o responsabili, affidabili, consultabili e informate (RACI).
    b. Definizioni dei team o pagine wiki.
    c. Definizioni e offerte di servizi.
    d. Ruolo o descrizione delle mansioni lavorative.
2. Esamina la documentazione e organizza discussioni sulle responsabilità documentate:
    a. Collaborando con i team identifica i disallineamenti tra le responsabilità documentate e quelle normalmente assunte dai team.
    b. Esamina i potenziali servizi offerti dai clienti interni per identificare le lacune nelle aspettative tra i team.
3. Analizza e risolvi le discrepanze.
4. Identifica le opportunità di miglioramento.
    a. Identifica le richieste più frequenti e con uso intensivo di risorse, che in genere sono ottime candidate al miglioramento.
    b. Esamina le best practice, i modelli e le linee guida prescrittive per semplificare e standardizzare i miglioramenti.
    c. Registra le opportunità di miglioramento e monitorale fino al completamento.
5. Se nessuno nel team è responsabile della gestione e del monitoraggio dell'assegnazione delle responsabilità, identifica qualcuno che assuma tale responsabilità.
6. Definisci un processo per consentire ai team di richiedere chiarimenti sulla responsabilità.
    a. Esamina il processo e verifica che sia chiaro e semplice da usare.
    b. Assicurati che qualcuno sia proprietario e segua le escalation fino al completamento.
    c. Stabilisci le metriche operative per misurare l'efficacia.

   d. Crea un meccanismo di feedback per verificare che i team possano evidenziare le opportunità di miglioramento.
   e. Implementa un meccanismo di revisione periodica.
7. Rendi i documenti disponibili in una posizione individuabile e accessibile.
   a. I wiki o il portale di documentazione sono le posizioni normalmente scelte.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS01-BP06 Valutazione dei compromessi
- OPS03-BP02 Potere di intervento dei membri del team quando i risultati sono a rischio
- OPS03-BP03 Incoraggiamento all'escalation
- OPS03-BP07 Fornitura di risorse appropriate ai team
- OPS09-BP01 Misura gli obiettivi operativi e i KPI con le metriche
- OPS09-BP03 Revisione delle metriche operative e assegnazione delle priorità per favorire il miglioramento
- OPS11-BP01 Definizione di un processo per il miglioramento continuo

Documenti correlati:

- Whitepaper AWS - Introduction to DevOps on AWS
- AWS Whitepaper - Cloud AWS Adoption Framework: Operations Perspective
- AWS Well-Architected Framework Operational Excellence - Workload level Operating model topologies
- AWS Prescriptive Guidance - Building your Cloud Operating Model
- AWS Prescriptive Guidance - Create a RACI or RASCI matrix for a cloud operating model
- Cloud AWS Operations & Migrations Blog - Delivering Business Value with Cloud Platform Teams
- Cloud AWS Operations & Migrations Blog - Why a Cloud Operating Model?
- AWS DevOps Blog - How organizations are modernizing for cloud operations

Video correlati:

- AWS Summit Online - Cloud Operating Models for Accelerated Transformation
- AWS re:Invent 2023 - Future-proofing cloud security: A new operating model

OPS02-BP05 Definizione di meccanismi per richiedere aggiunte, modifiche ed eccezioni

È possibile effettuare richieste ai proprietari di processi, procedure e risorse. Tra le richieste figurano aggiunte, modifiche ed eccezioni. Tali richieste passano attraverso un processo di gestione delle modifiche Prendi decisioni informate per approvare le richieste quando vengono ritenute fattibili e appropriate dopo una valutazione dei vantaggi e dei rischi.

Risultato desiderato:

- Puoi effettuare richieste per modificare processi, procedure e risorse sulla base della titolarità assegnata.
- Le modifiche vengono eseguite in modo deliberato, valutando benefici e rischi.

Anti-pattern comuni:

- Devi aggiornare il modo di implementare la tua applicazione, ma non esiste un metodo per richiedere una modifica al processo di implementazione al team operativo.
- Il piano di ripristino di emergenza deve essere aggiornato, ma non è stato identificato il proprietario a cui richiedere le modifiche.

Vantaggi dell'adozione di questa best practice:

- Processi, procedure e risorse possono evolvere mentre cambiano i requisiti.
- I titolati possono prendere decisioni mirate su quando effettuare le modifiche.
- Le modifiche vengono eseguite deliberatamente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Per implementare questa best practice devi essere in grado di richiedere modifiche a processi, procedure e risorse. Il processo di gestione delle modifiche può essere semplice. Documenta il processo di gestione delle modifiche.

Esempio del cliente

AnyCompany Retail usa una matrice di assegnazione delle responsabilità (RACI) per identificare il proprietario delle modifiche per processi, procedure e risorse. L'azienda dispone di un processo documentato di gestione delle modifiche, semplice e facile da seguire. Tramite il processo e la matrice RACI, tutti possono inviare richieste di modifiche.

Passaggi dell'implementazione

1. Identifica i processi, le procedure e le risorse per il tuo carico di lavoro e i proprietari di ciascun elemento. Documentali nel tuo sistema di gestione delle conoscenze.
   a. Se non hai implementato OPS02-BP01 Associazione di proprietari identificati alle risorse, OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure o OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni, comincia per prima cosa da questi.
2. Collabora con le parti interessate all'interno della tua azienda per sviluppare un processo di gestione delle modifiche. Il processo deve includere aggiunte, modifiche ed eccezioni per risorse, processi e procedure.
   a. Puoi utilizzare AWS Systems Manager Change Manager come piattaforma di gestione delle modifiche per le risorse dei carichi di lavoro.
3. Documenta il processo di gestione delle modifiche nel tuo sistema di gestione delle conoscenze.

Livello di impegno per il piano di implementazione: Medio. Sviluppare un processo di gestione delle modifiche significa garantire un allineamento con più parti interessate all'interno dell'organizzazione.

Risorse

Best practice correlate:

- OPS02-BP01 Associazione di proprietari identificati alle risorse - I titolari delle risorse devono essere identificati prima di definire un processo di gestione delle modifiche.
- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure - I titolari dei processi devono essere identificati prima di definire un processo di gestione delle modifiche.
- OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni - I titolari delle attività operative devono essere identificati prima di definire un processo di gestione delle modifiche.

Documenti correlati:

- Prontuario AWS - Playbook di base per grandi migrazioni AWS: creazione di matrici RACI
- Whitepaper sulla Gestione delle modifiche nel cloud

Servizi correlati:

- AWS Systems Manager Change Manager

OPS02-BP06 Predefinizione o negoziazione delle responsabilità tra i team

Fai in modo che esistano accordi definiti o negoziati tra i team che descrivono come funzionano e si supportano reciprocamente (ad esempio, tempi di risposta, obiettivi o contratti relativi al livello di servizio). I canali di comunicazione tra team sono documentati. Comprendere l'impatto del lavoro dei team sui risultati aziendali e sui risultati di altri team e organizzazioni fornisce indicazioni in merito alla priorità dei loro compiti e consente loro di rispondere in modo appropriato.

Quando la responsabilità e la proprietà sono indefinite o sconosciute, rischi sia di non affrontare le attività necessarie in modo tempestivo sia di impiegare sforzi ridondanti e potenzialmente conflittuali per rispondere a tali esigenze.

Risultato desiderato:

- Il lavoro tra team o gli accordi di assistenza vengono concordati e documentati.
- I team che supportano o lavorano con altri hanno definito i canali di comunicazione e le aspettative in termini di risposte.

Anti-pattern comuni:

- In produzione si verifica un problema e due team separati iniziano a cercare la soluzione senza confrontarsi. Il loro impegno separato prolunga l'interruzione.
- Il team operativo ha bisogno di assistenza dal team di sviluppo, ma non c'è un accordo sui tempi di risposta. La richiesta si blocca nel backlog.

Vantaggi dell'adozione di questa best practice:

- I team sanno come interagire e supportarsi a vicenda.

- Le aspettative relative ai tempi di risposta sono note.
- I canali di comunicazione sono definiti in modo chiaro.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Se si implementa questa best practice non ci saranno dubbi sulla collaborazione tra team. Gli accordi formali codificano il modo di collaborare o di assistersi a vicenda dei team. I canali di comunicazione tra team sono documentati.

Esempio del cliente

Il team SRE di AnyCompany Retail ha un contratto sul livello di servizio (SLA) con il team di sviluppo. Ogni volta che il team di sviluppo effettua una richiesta nel sistema di ticketing, riceverà una risposta entro 15 minuti. Se si verifica un malfunzionamento presso la sede, il team SRE assume il comando delle indagini con il supporto del team di sviluppo.

Passaggi dell'implementazione

1. Collaborando con le parti interessate all'interno dell'organizzazione, sviluppa accordi tra team basati su processi e procedure.
   a. Se i due team condividono un processo o una procedura, crea un runbook su come i team devono collaborare.
   b. Se esistono dipendenze tra i team, concorda uno SLA per le risposte alle richieste.
2. Inserisci le responsabilità nel tuo sistema di gestione delle conoscenze.

Livello di impegno per il piano di implementazione: Medio. Se non esistono accordi tra i team, può essere impegnativo raggiungere un accordo con le parti interessate all'interno dell'organizzazione.

Risorse

Best practice correlate:

- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure - La titolarità dei processi deve essere stabilita prima di definire gli accordi tra i team.
- OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni - La titolarità delle attività operative deve essere stabilita prima di definire gli accordi tra i team.

Documenti correlati:

- AWS Executive Insights - Promuovere l'innovazione con i team da due pizze
- Introduzione a DevOps su AWS - I team da due pizze

OPS 3. In che modo la cultura aziendale supporta i risultati aziendali?

Fornisci supporto ai membri del team in modo che possano essere più efficaci nell'azione e nel supporto dei risultati aziendali.

Best practice

- OPS03-BP01 Definizione della sponsorizzazione esecutiva
- OPS03-BP02 Potere di intervento dei membri del team quando i risultati sono a rischio
- OPS03-BP03 Incoraggiamento all'escalation
- OPS03-BP04 Comunicazioni tempestive, chiare e fruibili
- OPS03-BP05 Incoraggiamento alla sperimentazione
- OPS03-BP06 Incoraggiamento ai membri del team a mantenere e ampliare le proprie competenze
- OPS03-BP07 Fornitura di risorse appropriate ai team

OPS03-BP01 Definizione della sponsorizzazione esecutiva

Ai massimi livelli, gli alti dirigenti fungono da sponsor esecutivo per definire chiaramente le aspettative e la direzione dei risultati dell'organizzazione, compresa la valutazione del successo. Lo sponsor sostiene e promuove l'adozione delle best practice e l'evoluzione dell'organizzazione.

Risultato desiderato: le organizzazioni che si impegnano ad adottare, trasformare e ottimizzare le operazioni cloud stabiliscono chiare linee di leadership e responsabilità per i risultati desiderati. L'organizzazione comprende ogni capacità richiesta per raggiungere un nuovo risultato e assegna la titolarità ai team funzionali per lo sviluppo. La leadership implementa attivamente questa direzione, assegna la titolarità, si assume la responsabilità e definisce il lavoro. Di conseguenza, le persone in tutta l'organizzazione possono mobilitarsi, sentirsi ispirate e lavorare attivamente per raggiungere gli obiettivi desiderati.

Anti-pattern comuni:

- I titolari dei carichi di lavoro sono tenuti a migrare i carichi di lavoro su AWS senza uno sponsor e un piano chiari per le operazioni cloud. I team pertanto non collaborano consapevolmente per

migliorare e consolidare le proprie capacità operative. La mancanza di standard operativi sulle best practice mette in difficoltà i team, ad esempio il lavoro degli operatori, le chiamate e il debito tecnico, limitando l'innovazione.

- È stato fissato un nuovo obiettivo a livello di organizzazione per adottare una tecnologia emergente senza fornire sponsor e strategia di leadership. I team interpretano gli obiettivi in modo diverso, il che crea confusione su dove concentrare gli impegni, sul perché sono importanti e su come misurare l'impatto. Di conseguenza, l'organizzazione perde slancio nell'adozione della tecnologia.

Vantaggi dell'adozione di questa best practice: quando lo sponsor esecutivo comunica e condivide chiaramente visione, direzione e obiettivi, i membri del team sanno cosa devono fare. Quando i leader sono coinvolti attivamente, le persone e i team iniziano a concentrare attivamente gli impegni nella stessa direzione per raggiungere gli obiettivi definiti. L'organizzazione di conseguenza massimizza la capacità di successo. Quando si valuta il successo, è possibile identificare meglio gli ostacoli al suo conseguimento in modo che possano essere affrontati attraverso l'intervento dello sponsor esecutivo.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

- In ogni fase del percorso verso il cloud (migrazione, adozione oppure ottimizzazione), il successo richiede un coinvolgimento attivo ai massimi livelli della leadership con uno sponsor esecutivo designato. Lo sponsor esecutivo allinea la mentalità, le competenze e le modalità di lavoro del team alla strategia definita.
  - Spiega il perché: fai chiarezza e spiega il ragionamento alla base della visione e della strategia.
  - Stabilisci le aspettative: definisci e pubblica gli obiettivi delle organizzazioni, incluso il modo in cui vengono misurati i progressi e il successo.
  - Monitora il raggiungimento degli obiettivi: misura periodicamente il raggiungimento incrementale degli obiettivi, non solo il completamento delle attività. Condividi i risultati in modo da poter intraprendere le azioni appropriate se si evidenziano dei rischi.
  - Fornisci le risorse necessarie per raggiungere gli obiettivi: favorisci la collaborazione tra persone e team per creare le soluzioni giuste che portino ai risultati definiti. Ciò riduce o elimina gli attriti organizzativi.
  - Sostegno ai team: mantieni un coinvolgimento attivo con i tuoi team in modo da comprendere come stanno e se ci sono fattori esterni che li influenzano. Individua gli ostacoli che impediscono l'avanzamento dei team. Agisci per conto dei tuoi team per superare gli ostacoli e rimuovere gli

oneri superflui. Quando i team sono influenzati da fattori esterni, rivaluta gli obiettivi e modifica i target in base alle esigenze.

- Promuovi l'adozione delle best practice: riconosci le best practice che offrono vantaggi quantificabili e identifica creatori e destinatari. Incoraggia ulteriormente l'adozione per amplificare i vantaggi ottenuti.
- Incoraggia l'evoluzione dei team: crea una cultura del miglioramento continuo e apprendi in modo proattivo dai progressi compiuti e dagli errori. Incoraggia la crescita e lo sviluppo sia personale sia organizzativo. Usa dati e aneddoti per migliorare la visione e la strategia.

Esempio del cliente

AnyCompany Retail è in fase di trasformazione aziendale attraverso il rapido rinnovamento delle esperienze dei clienti, il miglioramento della produttività e l'accelerazione della crescita con l'IA generativa.

Passaggi dell'implementazione

1. Stabilisci una leadership a thread singolo e assegna uno sponsor esecutivo principale per guidare e gestire la trasformazione.
2. Definisci chiaramente i risultati aziendali della trasformazione e assegna titolarità e responsabilità. Fornisci allo sponsor esecutivo principale l'autorità di guidare e prendere decisioni critiche.
3. Verifica che la strategia di trasformazione sia stata definita molto chiaramente e sia stata ampiamente comunicata dallo sponsor esecutivo a tutti i livelli dell'organizzazione.
   a. Definisci chiaramente gli obiettivi aziendali per le iniziative IT e cloud.
   b. Documenta le principali metriche aziendali per promuovere la trasformazione dell'IT e del cloud.
   c. Comunica la visione in modo coerente a tutti i team e alle persone responsabili di parti della strategia.
4. Sviluppa matrici di pianificazione della comunicazione che specifichino quale messaggio deve essere recapitato a leader, manager e singoli collaboratori specifici. Specifica la persona o il team che deve recapitare questo messaggio.
   a. Rispetta i piani di comunicazione in modo coerente e affidabile.
   b. Stabilisci e gestisci le aspettative attraverso eventi di persona su base regolare.
   c. Accetta il feedback sull'efficacia delle comunicazioni, quindi modifica le comunicazioni e pianifica di conseguenza.

   d. Pianifica gli eventi di comunicazione per comprendere in modo proattivo le sfide dei team e stabilire un ciclo di feedback coerente che consenta di correggere la direzione laddove necessario.

5. Coinvolgi attivamente ogni iniziativa dal punto di vista della leadership per verificare che tutti i team interessati comprendano i risultati di cui sono responsabili.
6. In ogni riunione sullo stato, gli sponsor esecutivi devono individuare gli ostacoli, esaminare metriche, aneddoti o feedback dei team nonché misurare i progressi verso il raggiungimento degli obiettivi.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS03-BP04 Comunicazioni tempestive, chiare e fruibili
- OPS11-BP01 Definizione di un processo per il miglioramento continuo
- OPS11-BP07 Revisione dei parametri delle operazioni

Documenti correlati:

- Untangling Your Organisational Hairball: Highly Aligned
- The Living Transformation: Pragmatically approaching changes
- Becoming a Future-Ready Enterprise
- 7 Pitfalls to Avoid When Building a CCOE
- Navigating the Cloud: Key Performance Indicators for Success

Video correlati:

- AWS re:Invent 2023: A leader's guide to generative AI: Using history to shape the future (SEG204)

Esempi correlati:

- Prosci: Primary Sponsor's Role & Importance

OPS03-BP02 Potere di intervento dei membri del team quando i risultati sono a rischio

Il comportamento culturale della responsabilità instillato dalla leadership fa sì che ogni dipendente si senta autorizzato ad agire per conto dell'intera azienda, al di là del proprio ambito definito da ruolo e responsabilità. I dipendenti possono intervenire per identificare in modo proattivo i rischi man mano che emergono e intraprendere le azioni appropriate. Tale cultura consente ai dipendenti di prendere decisioni di alto valore in quanto sono consapevoli della situazione.

Ad esempio, Amazon utilizza i principi di leadership come linee guida per favorire il comportamento desiderato dei dipendenti per andare gestire situazioni complesse, risolvere i problemi, affrontare i conflitti e intraprendere le azioni adeguate.

Risultato desiderato: la leadership ha stabilito una nuova cultura che consente a persone e team di prendere decisioni critiche, anche ai livelli più bassi dell'organizzazione, purché tali decisioni siano definite con autorizzazioni e meccanismi di sicurezza verificabili. L'errore non è una mancanza, i team imparano in modo iterativo a migliorare il processo decisionale e le risposte per affrontare situazioni simili in futuro. Se le azioni già intraprese portano a un miglioramento che può avvantaggiare altri team, occorre condividere in modo proattivo le conoscenze derivanti da tali azioni. La leadership misura i miglioramenti operativi e incentiva le persone e l'organizzazione all'adozione di tali modelli.

Anti-pattern comuni:

- Nell'organizzazione non esistono linee guida o meccanismi chiari su cosa fare quando viene identificato un rischio. Ad esempio, quando un dipendente nota un attacco di phishing, non lo segnala al team di sicurezza, con il risultato che gran parte dell'organizzazione è vittima dell'attacco, causando una violazione dei dati.
- I clienti si lamentano dell'indisponibilità del servizio, che deriva principalmente da implementazioni non riuscite. Il team SRE è responsabile dello strumento di implementazione e il rollback automatico per le implementazioni è nella roadmap a lungo termine. In un recente rollout dell'applicazione, uno degli ingegneri ha fornito una soluzione per automatizzare il ripristino dell'applicazione a una versione precedente. Sebbene la soluzione possa diventare il modello per i team SRE, altri team non la adottano perché non esiste un processo per monitorare i miglioramenti. L'organizzazione continua a ottenere implementazioni non corrette che hanno un impatto sui clienti e provocano ulteriore insoddisfazione.
- Per rispettare la conformità, il team di infosec supervisiona un processo consolidato per ruotare regolarmente le chiavi SSH condivise per conto degli operatori che si connettono alle loro istanze Linux Amazon EC2. I team di infosec impiegano diversi giorni per completare la rotazione delle

chiavi e la connessione alle istanze viene bloccata. Nessuno all'interno o all'esterno di infosec suggerisce di utilizzare altre opzioni su AWS per ottenere lo stesso risultato.

Vantaggi dell'adozione di questa best practice: decentralizzando l'autorità del potere decisionale e consentendo ai team di prendere le decisioni chiave, è possibile risolvere i problemi più rapidamente con tassi di successo sempre crescenti. Inoltre, i team iniziano a percepire un senso di appartenenza e gli errori sono accettabili. La sperimentazione diventa un pilastro culturale. I manager e i direttori non si sentono controllati in ogni aspetto del loro lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

1. Sviluppa una cultura in cui si prevede che possano verificarsi errori.
2. Definisci chiaramente titolarità e responsabilità per le varie aree funzionali all'interno dell'organizzazione.
3. Comunica la titolarità e la responsabilità a tutti in modo che le persone sappiano chi può facilitare le decisioni decentralizzate.
4. Stabilisci le decisioni definitive e reversibili per permettere alle persone di sapere quando è necessario eseguire l'escalation a livelli più alti di leadership.
5. Crea la consapevolezza organizzativa che tutti i dipendenti hanno la capacità di agire a vari livelli quando i risultati sono a rischio. Fornisci ai membri del team la documentazione sulla governance, i livelli di autorizzazione, gli strumenti e le opportunità per mettere in pratica le competenze necessarie e intervenire in modo efficace.
6. Offri ai membri del team l'opportunità di mettere in pratica le competenze necessarie per rispondere a varie decisioni. Una volta definiti i livelli decisionali, organizza delle giornate di gioco per verificare che tutti i singoli collaboratori comprendano e possano usare il processo.
   a. Fornisci ambienti sicuri alternativi in cui testare i processi e sottoporre i membri del team alla dovuta formazione.
   b. Riconosci e crea la consapevolezza che i membri del team hanno l'autorità di agire quando il risultato ha un livello di rischio prestabilito.
   c. Definisci l'autorità dei membri del team per intervenire assegnando le autorizzazioni e l'accesso ai carichi di lavoro e ai componenti supportati.
7. Offri ai team la possibilità di condividere le proprie conoscenze (successi e fallimenti operativi).

8. Consenti ai team di sfidare lo status quo e fornisci i meccanismi per monitorare e misurare i miglioramenti, nonché il loro impatto sull'organizzazione.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS01-BP06 Valutazione dei compromessi gestendo vantaggi e rischi
- OPS02-BP05 Definizione di meccanismi per identificare responsabilità e titolarità

Documenti correlati:

- AWS Blog Post | The agile enterprise
- AWS Blog Post | Measuring success : A paradox and a plan
- AWS Blog Post | Letting go : Enabling autonomy in teams
- Centralize or Decentralize?

Video correlati:

- re:Invent 2023 | How to not sabotage your transformation (SEG201)
- re:Invent 2021 | Amazon Builders' Library: Operational Excellence at Amazon
- Centralization vs. Decentralization

Esempi correlati:

- Using architectural decision records to streamline technical decision-making for a software development project

OPS03-BP03 Incoraggiamento all'escalation

I membri del team sono incoraggiati dalla leadership a segnalare problemi e preoccupazioni ai responsabili delle decisioni e alle parti interessate di alto livello se ritengono che i risultati desiderati siano a rischio e gli standard previsti non siano rispettati. Questa è una funzionalità della cultura dell'organizzazione ed è implementata a tutti i livelli. L'escalation deve essere eseguita in anticipo e di

frequente, in modo che i rischi possano essere identificati e limitati prima che provochino incidenti. La leadership non rimprovera le persone per aver effettuato l'escalation di un problema.

Risultato desiderato: le persone in tutta l'organizzazione si sentono a proprio agio nell'eseguire l'escalation dei problemi ai livelli di leadership immediati e più elevati. La leadership ha stabilito deliberatamente e consapevolmente l'aspettativa che i propri team si sentano tranquilli nell'eseguire l'escalation di qualsiasi problema. Esiste un meccanismo per eseguire l'escalation dei problemi a ogni livello dell'organizzazione. Quando un dipendente esegue l'escalation al proprio manager, insieme decidono il livello di impatto e se il problema debba essere ulteriormente scalato. Per iniziare l'escalation, i dipendenti sono tenuti a includere un piano di lavoro consigliato per risolvere il problema. Se la direzione non interviene tempestivamente, i dipendenti sono incoraggiati a inoltrare i problemi al massimo livello di leadership se ritengono fermamente che i rischi per l'organizzazione giustifichino l'escalation.

Anti-pattern comuni:

- I dirigenti non pongono domande approfondite durante la riunione sullo stato del programma di trasformazione del cloud per scoprire dove si verificano problemi e ostacoli. Solo le buone notizie vengono presentate nello stato. Il CIO ha chiarito che le piacciono solo le buone notizie, poiché qualsiasi sfida sollevata fa pensare che il programma non stia andando bene.
- Sei un ingegnere delle operazioni cloud e noti che il nuovo sistema di gestione delle conoscenze non è ampiamente adottato dai team applicativi. L'azienda ha investito un anno di tempo e diversi milioni di dollari per implementare questo nuovo sistema di gestione delle conoscenze, ma le persone continuano a creare i propri runbook localmente e a condividerli su una condivisione cloud aziendale, rendendo difficile il recupero delle conoscenze pertinenti ai carichi di lavoro supportati. Cerchi di portare questo aspetto all'attenzione della dirigenza perché l'uso coerente del sistema può migliorare l'efficienza operativa. Quando lo comunichi alla direttrice a capo dell'implementazione del sistema di gestione delle conoscenze, ti rimprovera perché mette in discussione l'investimento.
- Il team di infosec responsabile del potenziamento delle risorse di calcolo ha deciso di mettere in atto un processo che richiede l'esecuzione delle scansioni necessarie per garantire che le istanze EC2 siano completamente protette prima che il team di calcolo rilasci la risorsa per l'uso. Ciò ha comportato il ritardo di una settimana per l'implementazione delle risorse e il mancato rispetto dello SLA. Il team di calcolo non desidera inoltrare la questione al vicepresidente tramite cloud perché ciò mette in cattiva luce il vicepresidente della sicurezza delle informazioni.

Vantaggi dell'adozione di questa best practice:

I problemi complessi o critici vengono risolti prima che abbiano impatto sull'azienda. Si perde meno tempo. I rischi sono ridotti al minimo. I team diventano più proattivi e concentrati sui risultati della risoluzione dei problemi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

La volontà e la capacità di crescere liberamente a tutti i livelli dell'organizzazione sono la base organizzativa e culturale che deve essere sviluppata consapevolmente attraverso una formazione appropriata, le comunicazioni della leadership, la definizione delle aspettative e l'implementazione di meccanismi a tutti i livelli dell'organizzazione.

Passaggi dell'implementazione

1. Definisci policy, standard e aspettative per l'organizzazione.
    1. Garantisci un'ampia adozione e comprensione delle policy, delle aspettative e degli standard.
2. Incoraggia, forma e responsabilizza i lavoratori a eseguire un'escalation anticipata e frequente quando gli standard non vengono rispettati.
3. Riconosci a livello organizzativo che l'escalation anticipata e frequente è la best practice. Accetti che le escalation possono rivelarsi infondate e che è meglio avere l'opportunità di prevenire un incidente piuttosto che privarsi di quell'opportunità senza escalation.
    a. Crea un meccanismo per l'escalation, ad esempio un sistema Andon Cord.
    b. È opportuno disporre di procedure documentate che definiscano quando e come deve verificarsi l'escalation.
    c. Definisci la serie di persone in ordine di autorità cui è consentito intraprendere o approvare azioni, nonché le informazioni di contatto di ciascuna parte interessata.
4. Un'escalation deve continuare fino a quando il membro del team non è convinto che il rischio sia stato mitigato attraverso le azioni guidate dalla leadership.
    a. Le escalation devono includere:
        i. la descrizione della situazione e la natura del rischio;
        ii. le criticità della situazione;
        iii. chi o cosa è interessato;
        iv. il livello dell'impatto;
        v. l'urgenza in caso di impatto;
        vi. i rimedi suggeriti e i piani di mitigazione.

   b. Proteggi i dipendenti coinvolti nell'escalation. È necessario predisporre una policy che protegga i membri del team da eventuali penalità se si trovano a dover scavalcare una parte interessata o un responsabile delle decisioni non reattivo. Metti in atto dei meccanismi per identificare se ciò si verifica e rispondere in modo appropriato.
5. Incoraggia la cultura del miglioramento continuo e dei cicli di feedback in tutto ciò che l'organizzazione produce. I cicli di feedback fungono da piccole escalation per le persone responsabili e identificano le opportunità di miglioramento, anche quando l'escalation non è necessaria. La cultura del miglioramento continuo obbliga tutti a essere più proattivi.
6. La leadership deve periodicamente ribadire le policy, gli standard, i meccanismi e il desiderio di un'escalation aperta e di cicli di feedback continui senza penalità.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS02-BP05 Definizione di meccanismi per richiedere aggiunte, modifiche ed eccezioni

Documenti correlati:

- How do you foster a culture of continuous improvement and learning from Andon and escalation systems?
- The Andon Cord (IT Revolution)
- AWS DevOps Guidance | Establish clear escalation paths and encourage constructive disagreement

Video correlati:

- Jeff Bezos on how to make decisions (& increase velocity)
- Toyota Product System: Stopping Production, a Button, and an Andon Electric Board
- Andon Cord in LEAN Manufacturing

Esempi correlati:

- Working with escalation plans in Incident Manager

OPS03-BP04 Comunicazioni tempestive, chiare e fruibili

La leadership è responsabile della creazione di comunicazioni forti ed efficaci, soprattutto quando l'organizzazione adotta nuove strategie, tecnologie o modalità di lavoro. I leader devono stabilire le aspettative affinché tutto il personale lavori per raggiungere gli obiettivi aziendali. Elabora meccanismi di comunicazione che creino e mantengano la consapevolezza tra i team responsabili della gestione dei piani finanziati e sponsorizzati dalla leadership. Utilizza la diversità interorganizzativa e ascolta con attenzione i molteplici punti di vista. Usa questa prospettiva per incrementare l'innovazione, mettere in discussione le tue ipotesi e ridurre il rischio di bias confermativi. Favorisci l'inclusione, la diversità e l'accessibilità all'interno dei team per ottenere prospettive vantaggiose.

Risultato desiderato: l'organizzazione progetta strategie di comunicazione per affrontare l'impatto del cambiamento sull'organizzazione. I team sono informati e motivati a continuare a lavorare insieme anziché l'uno contro l'altro. Le persone comprendono quanto sia importante il proprio ruolo per raggiungere gli obiettivi stabiliti. L'e-mail è solo un meccanismo passivo per le comunicazioni e viene utilizzato di conseguenza. La direzione trascorre del tempo con i singoli collaboratori per aiutarli a comprendere le proprie responsabilità, le attività da completare e in che modo il loro lavoro contribuisce alla missione generale. Quando necessario, i leader coinvolgono direttamente le persone in un ambiente più piccolo per trasmettere il messaggio e verificare che venga recepito in modo efficace. Come risultato di buone strategie di comunicazione, l'organizzazione si comporta in misura pari o superiore alle aspettative della leadership. La leadership incoraggia e desidera esaminare opinioni diverse all'interno dell'organizzazione e tra i team.

Anti-pattern comuni:

- L'organizzazione ha un piano quinquennale per migrare tutti i carichi di lavoro su AWS. Il business case per il cloud include la modernizzazione del 25% di tutti i carichi di lavoro per utilizzare la tecnologia serverless. Il CIO comunica questa strategia ai collaboratori diretti e si aspetta che ogni leader trasmetta questa presentazione a manager, direttori e singoli collaboratori senza comunicazioni di persona. Il CIO fa un passo indietro e si aspetta che l'organizzazione esegua la nuova strategia.
- La leadership non fornisce né utilizza un meccanismo di feedback e aumenta il divario nelle aspettative, causando ostacoli per i progetti.
- Ti viene chiesto di apportare una modifica ai gruppi di sicurezza, ma non ti vengono forniti i dettagli sulle modifiche da apportare, sull'impatto della modifica su tutti i carichi di lavoro e sulla data della modifica. Il manager inoltra un'e-mail del vicepresidente di infosec e aggiunge il messaggio "Make this happen."

- Sono state apportate modifiche alla strategia di migrazione che riducono la percentuale di modernizzazione pianificata dal 25% al 10%. La riduzione ha effetti a valle sull'organizzazione delle operazioni. Questo cambiamento strategico non è stato comunicato e quindi non è disponibile la capacità qualificata sufficiente per supportare un numero maggiore di carichi di lavoro lift and shift AWS.

Vantaggi dell'adozione di questa best practice:

- L'organizzazione è ben informata sulle strategie nuove o modificate e agisce di conseguenza con una forte motivazione alla collaborazione per raggiungere gli obiettivi e le metriche generali stabiliti dalla leadership.
- Esistono meccanismi che vengono utilizzati per fornire tempestivamente notifiche ai membri del team in merito a rischi noti ed eventi pianificati.
- Le nuove modalità di lavoro, compresi i cambiamenti delle persone o dell'organizzazione, dei processi o della tecnologia, insieme alle competenze richieste, vengono adottate in modo più efficace dall'organizzazione che quindi realizza i vantaggi aziendali più rapidamente.
- I membri del team hanno il contesto necessario per ricevere le comunicazioni e possono essere più efficaci nel loro lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per implementare questa best practice devi collaborare con le parti interessate presenti nell'organizzazione per concordare gli standard di comunicazione. Comunica tali standard alla tua organizzazione. Per qualsiasi transizione IT significativa, il team di pianificazione definito può gestire con maggiore successo l'impatto del cambiamento sulle persone rispetto a un'organizzazione che ignora questa procedura. Le organizzazioni più grandi possono essere più impegnative nella gestione del cambiamento perché richiedono un forte consenso sulla nuova strategia di tutti i singoli collaboratori. In assenza di un team di pianificazione della transizione, la leadership ha il 100% della responsabilità di condurre comunicazioni efficaci. Quando si crea un team di pianificazione della transizione, comunica ai membri del team di collaborare con tutta la leadership organizzativa per definire e gestire comunicazioni efficaci a tutti i livelli.

Esempio del cliente

AnyCompany Retail si è registrata al supporto Enterprise AWS e dipende da altri fornitori di terze parti per le operazioni cloud. L'azienda utilizza chat e chatop come principale mezzo di comunicazione per le attività operative. Allarmi e altre informazioni popolano canali specifici. Quando qualcuno deve intervenire, il risultato desiderato viene definito in modo chiaro e, in molti casi, la persona riceve un runbook o un playbook da usare. Usa un calendario delle modifiche per pianificare i cambiamenti più importanti ai sistemi di produzione.

Passaggi dell'implementazione

1. Crea un team principale all'interno dell'organizzazione che abbia la responsabilità di elaborare e avviare i piani di comunicazione dei cambiamenti che avvengono a più livelli all'interno dell'organizzazione.
2. Istituisci la titolarità a thread singolo per la supervisione. Offri ai singoli team la capacità di innovare in modo indipendente e bilanciare l'uso di meccanismi coerenti, consentendo così il giusto livello di ispezione e visione della direzione.
3. Collabora con le parti interessate di tutta l'organizzazione per concordare standard, procedure e piani di comunicazione.
4. Verifica che il team di comunicazione principale collabori con la leadership dell'organizzazione e del programma per creare messaggi per il personale appropriato per conto dei leader.
5. Sviluppa meccanismi di comunicazione strategici per gestire il cambiamento attraverso annunci, calendari condivisi, riunioni plenarie e metodi di persona o individuali, in modo che i membri del team abbiano le giuste aspettative sulle azioni da intraprendere.
6. Quando possibile, fornisci il contesto, i dettagli e il tempo necessari per determinare se è richiesta un'azione. Quando è necessaria un'azione, fornisci l'azione richiesta e il suo impatto.
7. Implementa gli strumenti che facilitino le comunicazioni tattiche, come chat interna, e-mail e gestione delle conoscenze.
8. Implementa i meccanismi per misurare e verificare che tutte le comunicazioni portino ai risultati desiderati.
9. Stabilisci un ciclo di feedback che misuri l'efficacia delle comunicazioni, specialmente quando sono correlate alla resistenza ai cambiamenti nell'organizzazione.
10. Per tutti gli Account AWS, stabilisci contatti alternativi per la fatturazione, la sicurezza e le operazioni. Idealmente, ogni contatto deve essere una distribuzione di e-mail anziché una comunicazione individuale specifica.
11. Stabilisci un piano di comunicazione di escalation e annullamento dell'escalation per interagire con i team interni ed esterni, compreso il supporto AWS e altri fornitori di terze parti.

12. Avvia ed esegui le strategie di comunicazione in modo coerente per tutta la durata di ciascun programma di trasformazione.
13. Assegna le priorità alle azioni ripetibili, ove possibile, per automatizzarle in sicurezza su larga scala.
14. Quando le comunicazioni sono richieste in scenari con azioni automatizzate, lo scopo della comunicazione deve essere quello di informare i team, per il controllo o una parte del processo di gestione delle modifiche.
15. Analizza le comunicazioni provenienti dai sistemi di avviso per individuare i falsi positivi o gli avvisi che vengono creati costantemente. Rimuovi o modifica questi avvisi in modo che vengano inviati quando è richiesto l'intervento umano. Se viene attivato un avviso, fornisci un runbook o un playbook.
    a. Puoi usare AWS Systems Manager Documents per creare playbook e runbook per gli avvisi.
16. Sono stati attivati meccanismi per fornire tempestivamente notifiche in merito ai rischi o agli eventi pianificati in modo chiaro e fruibile al fine di consentire risposte appropriate. Usa elenchi di indirizzi e-mail o canali di chat per inviare le notifiche di preavviso rispetto agli eventi pianificati.
    a. AWS Chatbot può essere utilizzato per inviare allarmi e rispondere agli eventi all'interno della piattaforma di messaggistica dell'organizzazione.
17. Fornisci una fonte di informazioni accessibile dove è possibile individuare gli eventi pianificati. Fornisci le notifiche degli eventi pianificati dallo stesso sistema.
    a. Il calendario delle modifiche di AWS Systems Manager può essere usato per creare sessioni in cui possono verificarsi le variazioni. In questo modo i membri del team ricevono un preavviso su quando poter effettuare la modifica in modo sicuro.
18. Monitora le notifiche di vulnerabilità e le informazioni sulle patch per comprendere le vulnerabilità nell'ambiente sregolato e i potenziali rischi associati ai componenti del carico di lavoro. Invia notifiche ai membri del team in modo che possano intervenire.
    a. Puoi iscriverti ai Bollettini sulla sicurezza AWS per ricevere notifiche sulla vulnerabilità di AWS.
19. Cerca opinioni e prospettive diverse: incoraggia la condivisione dei contributi da parte di tutti. Offri opportunità di comunicazione ai gruppi sottorappresentati. Distribuisci a rotazione i ruoli e le responsabilità nelle riunioni.
    a. Amplia ruoli e responsabilità: offri ai membri del team l'opportunità di assumere ruoli che altrimenti potrebbero non ricoprire mai. Ciò consentirà loro di acquisire esperienza e nuove prospettive grazie anche alle interazioni con i nuovi membri del team, con i quali potrebbero non interagire altrimenti. Un mutuo scambio di esperienze e punti di vista vantaggioso per tutti. Con l'aumento delle prospettive, identifica le opportunità aziendali emergenti o le nuove opportunità

di miglioramento. Fai in modo che i membri di un team svolgano a turno attività comuni eseguite normalmente da altri affinché comprendano le richieste e l'impatto delle loro prestazioni.

b. Garantisci un ambiente sicuro e ospitale: adotta policy e controlli che consentano di proteggere la sicurezza fisica e mentale dei membri del team all'interno dell'organizzazione. I membri del team devono essere in grado di interagire senza alcun timore. Quando i membri del team si sentono al sicuro e ben accolti, è più probabile che siano coinvolti e produttivi. Più è diversificata la tua organizzazione, migliore sarà la comprensione nei confronti delle persone supportate, inclusi i clienti. Quando i membri del team si sentono a loro agio, sono liberi di parlare e sono sicuri che verranno ascoltati, con maggiori probabilità condivideranno approfondimenti preziosi (ad esempio, opportunità di marketing, esigenze di accessibilità, segmenti di mercato non serviti, rischi non riconosciuti nel tuo ambiente).

c. Incoraggia la totale partecipazione dei membri del team: fornisci le risorse necessarie ai dipendenti affinché partecipino appieno a tutte le attività correlate al lavoro. I membri del team che affrontano sfide quotidiane hanno sviluppato competenze per superarle. Queste competenze esclusive possono offrire vantaggi significativi alla tua organizzazione. Grazie al supporto di strutture adeguate, i membri del team possono apportare contributi vantaggiosi.

Risorse

Best practice correlate:

- OPS03-BP01 Definizione della sponsorizzazione esecutiva
- OPS07-BP03 Utilizzo di runbook per eseguire le procedure
- OPS07-BP04 Utilizzo dei playbook per analizzare i problemi

Documenti correlati:

- AWS Blog post | Accountability and empowerment are key to high-performing agile organizations
- AWS Executive Insights | Learn to scale innovation, not complexity | Single-threaded Leaders
- Bollettini sulla sicurezza AWS
- Open CVE
- AWS Support App in Slack to Manage Support Cases
- Manage AWS resources in your Slack channels with AWS Chatbot

Esempi correlati:

- Well-Architected Labs: Inventario e gestione delle patch (Livello 100)

Servizi correlati:

- AWS Chatbot
- AWS Systems Manager Change Calendar
- AWS Systems Manager Documents

OPS03-BP05 Incoraggiamento alla sperimentazione

La sperimentazione è un catalizzatore per trasformare nuove idee in prodotti e funzionalità. La sperimentazione accelera l'apprendimento e mantiene acceso l'interesse e il coinvolgimento dei membri del team. I membri del team sono incoraggiati a sperimentare spesso per promuovere l'innovazione. Anche quando si verifica un risultato indesiderato, è comunque utile sapere quello che non bisogna fare. I membri del team non vengono puniti per gli esperimenti riusciti con risultati indesiderati.

Risultato desiderato:

- La tua organizzazione incoraggia la sperimentazione per promuovere l'innovazione.
- Gli esperimenti sono utilizzati come un'opportunità per imparare.

Anti-pattern comuni:

- Vuoi eseguire un test A/B, ma non esiste un meccanismo per eseguire l'esperimento. Distribuisci una modifica all'interfaccia utente senza la possibilità di testarla. Questo comporta un'esperienza cliente negativa.
- La tua azienda ha solo un ambiente di test e uno di produzione. Non esiste un ambiente di sperimentazione (sandbox) in cui provare nuove funzionalità o prodotti, per cui le sperimentazioni vengono effettuate all'interno dell'ambiente di produzione.

Vantaggi dell'adozione di questa best practice:

- La sperimentazione incoraggia l'innovazione.
- Grazie alla sperimentazione puoi reagire più velocemente al feedback degli utenti.
- La tua organizzazione sviluppa una cultura dell'apprendimento.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Le sperimentazioni devono essere eseguite in modo sicuro. Sfrutta più ambienti per sperimentare senza mettere a rischio le risorse di produzione. Usa il test A/B e le flag delle funzionalità per testare gli esperimenti. Offri ai membri del team la possibilità di eseguire esperimenti in un ambiente di sperimentazione (sandbox).

Esempio del cliente

AnyCompany Retail incoraggia la sperimentazione. I membri del team possono dedicare il 20% della propria settimana lavorativa alla sperimentazione o all'apprendimento di nuove tecnologie. Hanno a disposizione un ambiente di sperimentazione (sandbox) in cui possono innovare. Il test A/B viene utilizzato per nuove funzionalità che possono essere così convalidate con il feedback di utenti reali.

Passaggi dell'implementazione

1. Lavora con la direzione della tua organizzazione per supportare la sperimentazione. I membri del team devono essere incoraggiati a eseguire esperimenti in modo sicuro.
2. Offri ai membri del team un ambiente in cui possono sperimentare in modo sicuro. Devono avere accesso a un ambiente simile alla produzione.
   a. Puoi usare un Account AWS separato per creare un ambiente di sperimentazione (sandbox). AWS Control Tower può essere usato per eseguire il provisioning di questi account.
3. Usa flag delle funzionalità e test A/B per sperimentare in modo sicuro e raccogliere il feedback degli utenti.
   a. AWS AppConfig Feature Flags offre la possibilità di creare flag delle funzionalità.
   b. Amazon CloudWatch Evidently può essere utilizzato per eseguire test A/B per un'implementazione limitata.
   c. Puoi usare le versioni AWS Lambda per implementare una nuova versione di una funzionalità per il test beta.

Livello di impegno per il piano di implementazione: alto Offrire ai membri del team un ambiente in cui sperimentare in modo sicuro può richiedere investimenti significativi. Potresti anche aver bisogno di modificare il codice dell'applicazione per usare flag di funzionalità o supportare il test A/B.

Risorse

Best practice correlate:

- OPS11-BP02 Esecuzione di analisi post-incidente - Imparare dagli incidenti è un fattore importante di innovazione e sperimentazione.
- OPS11-BP03 Implementazione di cicli di feedback - I cicli di feedback sono una parte importante della sperimentazione.

Documenti correlati:

- Uno sguardo approfondito alla cultura di Amazon: Sperimentazione, Fallimento e Ossessione per il cliente
- Best practice per creare e gestire account per un ambiente di sperimentazione (sandbox) in AWS
- Creare una cultura della sperimentazione abilitata dal Cloud
- Promuovere sperimentazione e innovazione nel cloud presso SulAmérica Seguros
- Sperimenta con più frequenza, sbaglia di meno
- Organizzazione dell'ambiente AWS con l'utilizzo di account multipli - OU Sandbox
- Usare AWS AppConfig Feature Flags

Video correlati:

- AWS On Air con Amazon CloudWatch Evidently | Eventi AWS
- AWS On Air San Fran Summit 2022 con integrazione di AWS AppConfig Feature Flags con Jira
- AWS re:Invent 2022 - Un'implementazione non è un rilascio: controlla i tuoi rilasci con flag di funzionalità (BOA305-R)
- Creazione programmatica di un Account AWS con AWS Control Tower
- Impostazione di un ambiente AWS multi-account che utilizzi le best practice di AWS Organizations

Esempi correlati:

- AWS Innovation Sandbox
- Personalizzazione end-to-end 101 per l'e-commerce

Servizi correlati:

- Amazon CloudWatch Evidently

- AWS AppConfig
- AWS Control Tower

OPS03-BP06 Incoraggiamento ai membri del team a mantenere e ampliare le proprie competenze

I team devono aumentare le proprie competenze per adottare nuove tecnologie e supportare i cambiamenti di domanda e responsabilità a supporto dei carichi di lavoro. L'ampliamento delle competenze nelle nuove tecnologie è spesso fonte di soddisfazione per i membri del team e supporta l'innovazione. Incoraggia i membri del team a perseguire e mantenere le certificazioni di settore in modo da convalidare e riconoscere le loro crescenti competenze. Pratica la formazione trasversale per promuovere il trasferimento di conoscenze e ridurre il rischio di impatto significativo in caso di perdita di membri del team qualificati ed esperti con competenze a livello istituzionale. Fornisci tempo strutturato dedicato per l'apprendimento.

AWS offre le risorse, tra cui il centro risorse per le nozioni di base di AWS, i blog AWS, i Tech talk online di AWS, gli eventi e i webinar di AWS e gli AWS Well-Architected Labs, che forniscono indicazioni, esempi e procedure guidate dettagliate per formare i team.

Risorse come AWS Support, (AWS re:Post, AWS Support Center) e AWS Documentation aiutano a rimuovere gli ostacoli tecnici e a migliorare le operazioni. Se hai domande, contatta AWS Support tramite AWS Support Center.

Inoltre, AWS condivide le best practice e i modelli appresi attraverso la gestione di AWS nella Amazon Builders' Library e un'ampia gamma di altri materiali didattici utili tramite il blog AWS e il podcast ufficiale di AWS.

AWS Training and Certification offre la formazione gratuita attraverso corsi digitali personalizzati, insieme a piani di apprendimento per ruolo o dominio. Per supportare ulteriormente lo sviluppo delle competenze AWS dei team, è anche possibile iscriversi a corsi di formazione con istruttore.

Risultato desiderato: l'organizzazione valuta costantemente le lacune nelle competenze e le colma con budget e investimenti strutturati. I team incoraggiano e incentivano i membri con attività di miglioramento delle competenze, come l'acquisizione delle principali certificazioni del settore. I team traggono beneficio da programmi dedicati alla condivisione incrociata delle conoscenze, come corsi di formazione in pausa pranzo, giornate di full immersion, hackathon e giornate di gioco. L'organizzazione mantiene i sistemi delle conoscenze aggiornati e pertinenti per la formazione incrociata dei membri dei team, compresi i corsi di formazione per l'onboarding dei nuovi assunti.

Anti-pattern comuni:

- In assenza di un programma di formazione strutturato e di un budget, i team riscontrano difficoltà nel tentativo di tenere il passo con l'evoluzione della tecnologia, il che si traduce in un aumento dell'attrito.
- Nell'ambito della migrazione ad AWS, l'organizzazione dimostra lacune nelle competenze e una padronanza del cloud variabile tra i team. Senza un impegno per il miglioramento delle competenze, i team si ritrovano oberati di attività di gestione legacy e inefficienti dell'ambiente cloud, causando un aumento del lavoro degli operatori. Questo stato di esaurimento dei team aumenta l'insoddisfazione dei dipendenti.

Vantaggi dell'adozione di questa best practice: quando un'organizzazione investe consapevolmente nel miglioramento delle competenze dei team, aiuta anche ad accelerare e scalare l'adozione e l'ottimizzazione del cloud. I programmi di apprendimento mirati favoriscono l'innovazione e creano capacità operative per consentire ai team di essere preparati a gestire gli eventi. I team investono consapevolmente nell'implementazione e nell'evoluzione delle best practice. Il morale dei team è alto e i membri apprezzano il contributo che offrono all'azienda.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Per adottare nuove tecnologie, promuovere l'innovazione e stare al passo con i cambiamenti della domanda e delle responsabilità a supporto dei carichi di lavoro, investi continuamente nella crescita professionale dei team.

Passaggi dell'implementazione

1. Utilizza programmi strutturati di sostegno per il cloud: AWS Skills Guild offre una formazione di consulenza per aumentare la fiducia nelle competenze cloud e promuovere la cultura dell'apprendimento continuo.
2. Metti a disposizione le risorse per la formazione: metti a disposizione tempo strutturato dedicato, accesso ai materiali di formazione, risorse di laboratorio e supporto alla partecipazione a conferenze e organizzazioni professionali che offrono opportunità di apprendimento da docenti e colleghi. Offri ai membri dei team junior la possibilità di contattare i membri dei team senior affinché questi fungano da mentori o possano mostrare loro come lavorano trasmettendo metodi e competenze consolidati. Incoraggia l'apprendimento dei contenuti non direttamente correlati al lavoro per avere una prospettiva più ampia.
3. Incoraggia l'uso di risorse tecniche esperte: utilizza risorse come AWS re:Post per accedere a conoscenze curate e a un'intraprendente community.

4. Crea e mantieni aggiornato l'archivio delle conoscenze: utilizza le piattaforme di condivisione delle conoscenze, come wiki e runbook. Crea la tua fonte di conoscenze specialistiche riutilizzabile con AWS re:Post Private per semplificare la collaborazione, migliorare la produttività e accelerare l'onboarding dei dipendenti.
5. Formazione del team e coinvolgimento tra team: pianifica le esigenze di formazione continua dei membri del tuo team. Offri loro l'opportunità di unirsi ad altri team (temporaneamente o definitivamente) per condividere competenze e best practice a beneficio dell'intera organizzazione.
6. Supporta il perseguimento e il mantenimento delle certificazioni di settore: favorisci l'acquisizione e il mantenimento da parte dei membri del tuo team di certificazioni di settore che convalidano le loro conoscenze e riconoscono i loro risultati.

Livello di impegno per il piano di deployment: elevato

Risorse

Best practice correlate:

- OPS03-BP01 Definizione della sponsorizzazione esecutiva
- OPS11-BP04 Gestione delle conoscenze

Documenti correlati:

- AWS Whitepaper | Cloud Adoption Framework: People Perspective
- Investing in continuous learning to grow your organization's future
- AWS Skills Guild
- AWS Training and Certification
- AWS Support
- AWS re:Post
- AWS Getting Started Resource Center
- AWS Blogs
- Conformità di Cloud AWS
- AWS Documentation
- The Official AWS Podcast.
- AWS Online Tech Talks

- Eventi e webinar AWS
- AWS Well-Architected Labs
- The Amazon Builders' Library

Video correlati:

- AWS re:Invent 2023 | Reskilling at the speed of cloud: Turning employees into entrepreneurs
- WS re:Invent 2023 | Building a culture of curiosity through gamification

OPS03-BP07 Fornitura di risorse appropriate ai team

Stabilisci il giusto numero di membri competenti del team e gli strumenti e le risorse per supportare le esigenze di carico di lavoro. Il sovraccarico dei membri del team aumenta il rischio di errore umano. Gli investimenti in strumenti e risorse, come l'automazione, possono aumentare l'efficacia del team consentendogli di supportare un numero maggiore di carichi di lavoro senza richiedere capacità aggiuntiva.

Risultato desiderato:

- Hai istituito un team con il personale necessario e le competenze richieste per gestire i carichi di lavoro in AWS in conformità al piano di migrazione. Man mano che il team si è ampliato nel corso del progetto di migrazione, ha acquisito competenza nelle tecnologie AWS di base che l'azienda intende utilizzare per la migrazione o la modernizzazione delle applicazioni.
- Hai preparato con attenzione il piano per i membri del team per fare un uso efficiente delle risorse, sfruttando l'automazione e il flusso di lavoro. Un team più piccolo può ora gestire più infrastrutture per conto dei team di sviluppo delle applicazioni.
- Con il cambiamento delle priorità operative, qualsiasi vincolo di risorse viene identificato in modo proattivo per proteggere il successo delle iniziative aziendali.
- Le metriche che segnalano le difficoltà operative, ad esempio l'affaticamento da chiamata o il paging eccessivo, vengono esaminate per verificare che il personale non sia sovraccaricato.

Anti-pattern comuni:

- Il personale non ha incrementato le competenze AWS all'approssimarsi del piano pluriennale di migrazione al cloud, comportando rischi per il supporto dei carichi di lavoro e abbassando il morale dei dipendenti.

- L'intera organizzazione IT adotta le modalità di lavoro agili. L'azienda assegna le priorità al portafoglio di prodotti e stabilisce le metriche per le funzionalità che devono essere sviluppate per prime. Il processo agile non richiede che i team assegnino story point ai piani di lavoro. Di conseguenza, è impossibile conoscere il livello di capacità richiesto per il successivo lavoro o se le competenze giuste sono state assegnate al lavoro.
- Un partner AWS migra i tuoi carichi di lavoro e non hai un piano di transizione di supporto per i team una volta completato il progetto di migrazione. I team hanno difficoltà a supportare i carichi di lavoro in modo efficiente ed efficace.

Vantaggi dell'adozione di questa best practice: nell'organizzazione sono disponibili membri dei team adeguatamente qualificati per supportare i carichi di lavoro. L'allocazione delle risorse può adattarsi al cambiamento delle priorità senza influire sulle prestazioni. Il risultato è che i team sono in grado di supportare i carichi di lavoro massimizzando al contempo il tempo per concentrarsi sull'innovazione per i clienti e aumentando a sua volta la soddisfazione dei dipendenti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La pianificazione delle risorse per la migrazione al cloud deve avvenire a un livello organizzativo in linea con il piano di migrazione e l'implementazione del modello operativo desiderato per supportare il nuovo ambiente cloud. Ciò deve includere la comprensione delle tecnologie cloud utilizzate per i team di sviluppo aziendale e delle applicazioni. La leadership dell'infrastruttura e delle operazioni deve pianificare l'analisi del divario delle competenze, la formazione e la definizione dei ruoli per gli ingegneri che guidano l'adozione del cloud.

Passaggi dell'implementazione

1. Definisci i criteri per il successo dei team con metriche operative pertinenti, come la produttività del personale (ad esempio, i costi di supporto di un carico di lavoro o le ore spese dall'operatore per gli incidenti).
2. Definisci i meccanismi di pianificazione e ispezione della capacità delle risorse per verificare che il giusto equilibrio di capacità qualificata sia disponibile quando necessario e possa essere modificato nel tempo.
3. Crea i meccanismi, ad esempio inviando un sondaggio mensile ai team, per comprendere le sfide legate al lavoro che hanno un impatto sui team, come l'aumento delle responsabilità, i cambiamenti nella tecnologia, la mancanza di personale o l'aumento dei clienti supportati.

4. Utilizza questi meccanismi per interagire con i team e individuare le tendenze che possono contribuire alle sfide relative alla produttività dei dipendenti. Quando i team sono influenzati da fattori esterni, rivaluta gli obiettivi e modifica i target in base alle esigenze. Individua gli ostacoli che impediscono l'avanzamento dei team.
5. Verifica regolarmente se le risorse attualmente messe a disposizione sono ancora sufficienti o se devono essere aggiunte e apporta le modifiche appropriate ai team di supporto.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS03-BP06 Incoraggiamento ai membri del team a mantenere e ampliare le proprie competenze
- OPS09-BP03 Revisione delle metriche operative e assegnazione delle priorità per favorire il miglioramento
- OPS10-BP01 Utilizzo di un processo per la gestione di eventi, incidenti e problemi
- OPS10-BP07 Automazione delle risposte agli eventi

Documenti correlati:

- Cloud AWS Adoption Framework: People Perspective
- Becoming a Future-Ready Enterprise
- Prioritize your Employees' Skills to Drive Business Growth
- High performing organization - the Amazon Two-Pizza team
- How Cloud-Mature Enterprises Succeed

Preparazione

Domande

- OPS 4. Come si implementa l'osservabilità nel carico di lavoro?
- OPS 5. In che modo riduci i difetti, favorisci la correzione e migliori il flusso nella produzione?
- OPS 6. In che modo mitighi i rischi della distribuzione?
- OPS 7. Come fai a sapere se hai tutto pronto per supportare un carico di lavoro?

OPS 4. Come si implementa l'osservabilità nel carico di lavoro?

Implementare l'osservabilità nel carico di lavoro ti permette di comprendere lo stato di quest'ultimo e di adottare decisioni basate sui dati e che riflettono i requisiti aziendali.

Best practice

- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS04-BP02 Implementazione della telemetria dell'applicazione
- OPS04-BP03 Implementazione della telemetria dell'esperienza utente
- OPS04-BP04 Implementazione della telemetria delle dipendenze
- OPS04-BP05 Implementazione del tracciamento distribuito

OPS04-BP01 Identificazione degli indicatori chiave di prestazione

L'implementazione dell'osservabilità nel carico di lavoro inizia con la comprensione del suo stato e l'adozione di decisioni basate sui dati che riflettono i requisiti aziendali. Uno dei modi più efficaci per garantire l'allineamento tra le attività di monitoraggio e gli obiettivi aziendali è definire e monitorare gli indicatori chiave di prestazione (KPI).

Risultato desiderato: Pratiche di osservabilità efficienti e strettamente allineate agli obiettivi aziendali garantiscono che le attività di monitoraggio siano sempre al servizio di risultati aziendali tangibili.

Anti-pattern comuni:

- KPI non definiti: lavorare senza KPI chiari può portare ad attività di monitoraggio eccessive o insufficienti e alla perdita di segnali vitali.
- KPI statici: non riesaminare od ottimizzare i KPI man mano che il carico di lavoro o gli obiettivi aziendali si evolvono.
- Disallineamento: concentrarsi su metriche tecniche non direttamente correlate ai risultati aziendali o che sono più difficili da correlare ai problemi del mondo reale.

Vantaggi dell'adozione di questa best practice:

- Facilità di identificazione dei problemi: i KPI aziendali spesso evidenziano i problemi in modo più chiaro rispetto alle metriche tecniche. Un valore di un KPI aziendale che diminuisce permette di individuare un problema in modo più efficace rispetto alla valutazione di numerose metriche tecniche.

- Allineamento aziendale: assicura che le attività di monitoraggio supportino direttamente gli obiettivi aziendali.
- Efficienza: viene data la priorità alle risorse di monitoraggio e al focus sulle metriche che contano.
- Proattività: riconoscere e risolvere i problemi prima che abbiano implicazioni aziendali più ampie.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Per definire in modo efficace i KPI del carico di lavoro:

1. Inizia con i risultati aziendali: prima di approfondire le metriche, comprendi i risultati aziendali desiderati. È stato rilevato un aumento delle vendite, un maggiore coinvolgimento degli utenti o tempi di risposta più rapidi?
2. Correla le metriche tecniche con gli obiettivi aziendali: non tutte le metriche tecniche hanno un impatto diretto sui risultati aziendali. Identifica quelli che hanno un impatto, anche se spesso è più immediato individuare un problema utilizzando un KPI aziendale.
3. Utilizza Amazon CloudWatch: Utilizza CloudWatch per definire e monitorare le metriche che rappresentano i tuoi KPI.
4. Rivedi e aggiorna regolarmente i KPI: man mano che il carico di lavoro e la tua attività si evolvono, mantieni la pertinenza dei tuoi KPI.
5. Coinvolgi gli stakeholder: coinvolgi i team tecnici e aziendali nella definizione e revisione dei KPI.

Livello di impegno per il piano di implementazione: Medio

Risorse

Best practice correlate:

- the section called “OPS04-BP02 Implementazione della telemetria dell'applicazione”
- the section called “OPS04-BP03 Implementazione della telemetria dell'esperienza utente”
- the section called “OPS04-BP04 Implementazione della telemetria delle dipendenze”
- the section called “OPS04-BP05 Implementazione del tracciamento distribuito”

Documenti correlati:

- AWS Observability Best Practices

- CloudWatch User Guide
- AWS Observability Skill Builder Course

Video correlati:

- Developing an observability strategy

Esempi correlati:

- One Observability Workshop

OPS04-BP02 Implementazione della telemetria dell'applicazione

La telemetria dell'applicazione è la base su cui si fonda l'osservabilità del carico di lavoro. È fondamentale emettere dati di telemetria che offrano approfondimenti utili sullo stato dell'applicazione e sul raggiungimento degli obiettivi sia tecnici sia aziendali. Dalla risoluzione dei problemi alla misurazione dell'impatto di una nuova funzionalità fino all'allineamento con gli indicatori di prestazione chiave (KPI), la telemetria dell'applicazione garantisce informazioni su cui basare la creazione, il funzionamento e l'evoluzione del carico di lavoro.

Metriche, log e tracce costituiscono i tre pilastri principali dell'osservabilità. Questi operano come strumenti diagnostici che descrivono lo stato dell'applicazione. Nel tempo, aiutano a creare criteri di base e a identificare le anomalie. Tuttavia, per garantire l'allineamento tra le attività di monitoraggio e gli obiettivi aziendali, è fondamentale definire e monitorare i KPI. I KPI aziendali spesso facilitano l'identificazione dei problemi rispetto alle sole metriche tecniche.

Altri tipi di telemetria, come il monitoraggio degli utenti reali (RUM) e le transazioni sintetiche, completano queste origini dati primarie. Il RUM offre approfondimenti sulle interazioni degli utenti in tempo reale, mentre le transazioni sintetiche simulano i potenziali comportamenti degli utenti, aiutando a rilevare i colli di bottiglia prima che vengano riscontrati dagli utenti reali.

Risultato desiderato: ottieni approfondimenti utili sulle prestazioni del tuo carico di lavoro. Questi approfondimenti consentono di prendere decisioni proattive sull'ottimizzazione delle prestazioni, ottenere una maggiore stabilità del carico di lavoro, semplificare i processi CI/CD e utilizzare le risorse in modo efficace.

Anti-pattern comuni:

- Osservabilità incompleta: si trascura di incorporare l'osservabilità a ogni livello del carico di lavoro, con conseguenti punti ciechi che possono nascondere le prestazioni vitali del sistema e gli approfondimenti sul comportamento.
- Visualizzazione frammentata dei dati: quando i dati sono sparsi su più strumenti e sistemi, diventa difficile mantenere una visione olistica dello stato e delle prestazioni del carico di lavoro.
- Problemi segnalati dagli utenti: un segno della mancanza di un rilevamento proattivo dei problemi tramite telemetria e monitoraggio dei KPI aziendali.

Vantaggi dell'adozione di questa best practice:

- Processo decisionale informato: con gli approfondimenti ricavati dalla telemetria e dai KPI aziendali, puoi prendere decisioni basate sui dati.
- Migliore efficienza operativa: l'utilizzo delle risorse basato sui dati porta a un miglioramento dell'efficienza risparmiando sui costi.
- Maggiore stabilità del carico di lavoro: rilevamento e risoluzione più rapidi dei problemi con conseguente aumento dei tempi di attività.
- Processi CI/CD semplificati: gli approfondimenti ricavati dai dati di telemetria facilitano il perfezionamento dei processi e la distribuzione affidabile del codice.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per implementare la telemetria delle applicazioni per il carico di lavoro, utilizza i servizi AWS come Amazon CloudWatch e AWS X-Ray. Amazon CloudWatch fornisce una suite completa di strumenti di monitoraggio, che consente di osservare le risorse e le applicazioni in ambienti AWS e on-premises. Raccoglie, tiene traccia e analizza le metriche, consolida e monitora i dati di log e risponde alle modifiche che interessano le risorse, migliorando la comprensione del funzionamento del carico di lavoro. Integrato con altri servizi, AWS X-Ray consente di tenere traccia, analizzare ed eseguire il debug delle applicazioni, offrendoti una comprensione approfondita del comportamento del tuo carico di lavoro. Grazie a funzionalità come mappe dei servizi, distribuzioni di latenza e tempistiche di tracciamento, AWS X-Ray fornisce approfondimenti sulle prestazioni del carico di lavoro e sui colli di bottiglia che lo interessano.

Passaggi dell'implementazione

1. Identifica quali dati raccogliere: definisci le metriche, i log e le tracce essenziali che potrebbero offrire importanti approfondimenti sullo stato, le prestazioni e il comportamento del tuo carico di lavoro.
2. Implementa l'agente CloudWatch: l'agente CloudWatch è fondamentale nel fornire metriche di sistema e dell'applicazione e log dal carico di lavoro e dall'infrastruttura sottostante. L'agente CloudWatch può essere utilizzato anche per raccogliere tracce OpenTelemetry o X-Ray e inviarle a X-Ray.
3. Implementa il rilevamento delle anomalie per log e metriche: utilizza il rilevamento delle anomalie CloudWatch Logs e il rilevamento delle anomalie delle metriche CloudWatch per identificare automaticamente attività insolite nelle operazioni dell'applicazione. Questi strumenti utilizzano algoritmi di machine learning per rilevare e comunicare le anomalie, migliorando le capacità di monitoraggio e accelerando i tempi di risposta a potenziali interruzioni o minacce alla sicurezza. Configura queste funzionalità per gestire in modo proattivo lo stato e la sicurezza delle applicazioni.
4. Proteggi i dati di log sensibili: utilizza la protezione dei dati Amazon CloudWatch Logs per mascherare le informazioni sensibili all'interno dei log. Questa funzionalità aiuta a mantenere la privacy e la conformità con il rilevamento e il mascheramento automatici dei dati sensibili prima dell'accesso. Implementa il mascheramento dei dati per gestire e proteggere in modo sicuro i dettagli sensibili come le informazioni di identificazione personale (PII).
5. Definisci e monitora i KPI aziendali: stabilisci metriche personalizzate in linea con i risultati aziendali.
6. Strumenta la tua applicazione con AWS X-Ray: oltre a implementare l'agente CloudWatch, è fondamentale strumentare l'applicazione per generare dati di tracciamento. Questo processo può fornire ulteriori approfondimenti sul comportamento e sulle prestazioni del carico di lavoro.
7. Standardizza la raccolta dei dati nell'applicazione: standardizza le procedure di raccolta dei dati nell'applicazione. L'uniformità aiuta a correlare e analizzare i dati, fornendo una visione completa del comportamento dell'applicazione.
8. Implementa l'osservabilità su più account: migliora l'efficienza del monitoraggio su più account Account AWS con l'osservabilità su più account Amazon CloudWatch. Con questa funzionalità, puoi consolidare metriche, log e allarmi di diversi account in un'unica visualizzazione, semplificando la gestione e migliorando i tempi di risposta per i problemi identificati nell'ambiente AWS dell'organizzazione.

9. Analizza e agisci sui dati: una volta in atto la raccolta e la normalizzazione dei dati, utilizza Amazon CloudWatch per l'analisi di metriche e log e AWS X-Ray per l'analisi delle tracce. Tale analisi può fornire approfondimenti cruciali sullo stato, le prestazioni e il comportamento del carico di lavoro, guidando il processo decisionale.

Livello di impegno per il piano di deployment: elevato

Risorse

Best practice correlate:

- OPS04-BP01 Definizione dei KPI del carico di lavoro
- OPS04-BP03 Implementazione della telemetria dell'attività degli utenti
- OPS04-BP04 Implementazione della telemetria delle dipendenze
- OPS04-BP05 Implementazione della tracciabilità delle transazioni

Documenti correlati:

- AWS Observability Best Practices
- CloudWatch User Guide
- AWS X-Ray Developer Guide
- Strumentazione di sistemi distribuiti per visibilità operativa
- AWS Observability Skill Builder Course
- Quali sono le novità di Amazon CloudWatch
- Quali sono le novità di AWS X-Ray

Video correlati:

- AWS re:Invent 2022 - Observability best practices at Amazon
- AWS re:Invent 2022 - Developing an observability strategy

Esempi correlati:

- One Observability Workshop
- AWS Solutions Library: Application Monitoring with Amazon CloudWatch

OPS04-BP03 Implementazione della telemetria dell'esperienza utente

Acquisire informazioni approfondite sulle esperienze dei clienti e sulle interazioni con la tua applicazione è fondamentale. Il monitoraggio dell'utente reale (RUM) e le transazioni sintetiche sono strumenti molto efficaci per questo scopo. RUM fornisce dati sulle interazioni degli utenti reali, garantendo una prospettiva non filtrata della soddisfazione degli utenti, mentre le transazioni sintetiche simulano le interazioni degli utenti, aiutando a rilevare potenziali problemi prima che essi abbiano un impatto sugli utenti reali.

Risultato desiderato: Una visione olistica dell'esperienza del cliente, il rilevamento proattivo dei problemi e l'ottimizzazione delle interazioni degli utenti per offrire esperienze digitali fluide.

Anti-pattern comuni:

- Applicazioni senza monitoraggio dell'utente reale (RUM):
  - rilevamento ritardato dei problemi: senza RUM, potresti non accorgerti di rallentamenti o problemi di prestazioni fino a quando non ricevi lamentele da parte degli utenti. Questo approccio reattivo può causare insoddisfazione nei clienti.
  - Mancanza di informazioni sull'esperienza utente: non utilizzare RUM significa perdere dati cruciali che mostrano come gli utenti reali interagiscono con l'applicazione, il che limita la tua capacità di ottimizzare l'esperienza utente.
- Applicazioni senza transazioni sintetiche:
  - Casi limite trascurati: le transazioni sintetiche consentono di testare percorsi e funzioni che potrebbero non essere utilizzati frequentemente dagli utenti tipici, ma che sono fondamentali per determinate funzioni aziendali. Senza di esse, questi percorsi potrebbero non funzionare correttamente e passare inosservati.
  - Verifica della presenza di problemi quando l'applicazione non viene utilizzata: i test sintetici regolari possono simulare situazioni in cui gli utenti reali non interagiscono attivamente con l'applicazione, garantendo che il sistema funzioni sempre correttamente.

Vantaggi dell'adozione di questa best practice:

- Rilevamento proattivo dei problemi: identifica e risolvi i problemi potenziali prima che abbiano un impatto sugli utenti reali.
- Esperienza utente ottimizzata: grazie al suo feedback continuo, RUM aiuta a perfezionare e migliorare l'esperienza utente complessiva.

- Informazioni approfondite sulle prestazioni del dispositivo e del browser: scopri come si comporta la tua applicazione in vari dispositivi e browser e implementa ulteriori ottimizzazioni.
- Flussi di lavoro aziendali convalidati: transazioni sintetiche regolari assicurano che le funzionalità principali e i percorsi critici siano operativi ed efficienti in maniera costante.
- Prestazioni delle applicazioni migliorate: sfrutta le informazioni approfondite raccolte dai dati degli utenti reali per migliorare la reattività e l'affidabilità delle applicazioni.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Per eseguire la telemetria delle attività degli utenti sfruttando RUM e le transazioni sintetiche, AWS offre servizi come Amazon CloudWatch RUM e Amazon CloudWatch Synthetics. Metriche, log e tracce, insieme ai dati sulle attività degli utenti, forniscono una visione completa dello stato operativo dell'applicazione e dell'esperienza utente.

Passaggi dell'implementazione

1. Implementa Amazon CloudWatch RUM: integra la tua applicazione con CloudWatch RUM per raccogliere, analizzare e presentare dati relativi agli utenti reali.
   a. Utilizza la libreria JavaScript CloudWatch RUM per integrare RUM con la tua applicazione.
   b. Configura dashboard per visualizzare e monitorare i dati relativi agli utenti reali.
2. Configura CloudWatch Synthetics: crea canary o routine con script che simulano le interazioni degli utenti con la tua applicazione.
   a. Definisci i flussi di lavoro e i percorsi critici delle applicazioni.
   b. Progetta canary utilizzando script di CloudWatch Synthetics per simulare le interazioni degli utenti per questi percorsi.
   c. Pianifica e monitora i canary affinché si attivino a intervalli specifici, in modo da garantire controlli costanti delle prestazioni.
3. Analizza e intervieni sui dati: Utilizza i dati provenienti da RUM e transazioni sintetiche per ottenere informazioni e adottare misure correttive quando vengono rilevate anomalie. Usa dashboard CloudWatch e allarmi per ottenere informazioni costanti.

Livello di impegno per il piano di implementazione: Medio

Risorse

Best practice correlate:

- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS04-BP02 Implementazione della telemetria dell'applicazione
- OPS04-BP04 Implementazione della telemetria delle dipendenze
- OPS04-BP05 Implementazione del tracciamento distribuito

Documenti correlati:

- Amazon CloudWatch RUM Guide
- Amazon CloudWatch Synthetics Guide

Video correlati:

- Optimize applications through end user insights with Amazon CloudWatch RUM
- AWS on Air ft. Real-User Monitoring for Amazon CloudWatch

Esempi correlati:

- One Observability Workshop
- Repository Git per client Web Amazon CloudWatch RUM
- Using Amazon CloudWatch Synthetics to measure page load time

OPS04-BP04 Implementazione della telemetria delle dipendenze

La telemetria delle dipendenze è essenziale per monitorare lo stato e le prestazioni dei servizi e dei componenti esterni su cui si basa il carico di lavoro. Fornisce preziosi approfondimenti su reperibilità, timeout e altri eventi critici correlati alle dipendenze come DNS, database o API di terze parti. Dotando l'applicazione di strumenti per generare metriche, log e tracce relative a queste dipendenze, acquisisci una comprensione più chiara dei potenziali colli di bottiglia, problemi di prestazioni o errori che potrebbero influire sul carico di lavoro.

Risultato desiderato: le dipendenze su cui si basa il carico di lavoro funzionano come previsto, consentendo di gestire i problemi in modo proattivo e garantendo prestazioni ottimali del carico di lavoro.

Anti-pattern comuni:

- Scarsa attenzione alle dipendenze esterne: il focus è rivolto esclusivamente alle metriche interne dell'applicazione, trascurando quelle legate alle dipendenze esterne.
- Mancanza di monitoraggio proattivo: si attende che si verifichino problemi anziché monitorare costantemente lo stato e le prestazioni delle dipendenze.
- Monitoraggio isolato in comparti: si utilizzano strumenti di monitoraggio multipli ed eterogenei che possono portare a visioni dello stato delle dipendenze frammentate e incoerenti.

Vantaggi dell'adozione di questa best practice:

- Maggiore affidabilità del carico di lavoro: viene garantito che le dipendenze esterne siano costantemente disponibili e funzionino in modo ottimale.
- Rilevamento e risoluzione dei problemi più rapidi: identificazione e risoluzione proattiva dei problemi relativi alle dipendenze prima che influiscano sul carico di lavoro.
- Visione completa: acquisizione di una visione olistica dei componenti interni ed esterni che influenzano lo stato del carico di lavoro.
- Scalabilità del carico di lavoro migliorata: grazie alla comprensione dei limiti di scalabilità e delle caratteristiche prestazionali delle dipendenze esterne.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Implementa la telemetria delle dipendenze iniziando con l'identificazione dei servizi, dell'infrastruttura e dei processi da cui dipende il carico di lavoro. Esegui una valutazione quantitativa delle condizioni ottimali nelle quali tali dipendenze funzionano come previsto e poi determina quali dati sono necessari per misurarle. Con queste informazioni, puoi creare dashboard e avvisi che forniscono approfondimenti ai tuoi team operativi sullo stato di tali dipendenze. Usa gli strumenti AWS per scoprire e quantificare gli impatti quando le dipendenze non riescono a fornire le prestazioni necessarie. Rivedi costantemente la tua strategia per tenere conto dei cambiamenti relativi a priorità, obiettivi e alle informazioni dettagliate acquisite.

Passaggi dell'implementazione

Per implementare efficacemente la telemetria delle dipendenze:

1. Identifica le dipendenze esterne: collabora con le parti interessate per individuare le dipendenze esterne sulle quali si basa il carico di lavoro. Le dipendenze esterne possono comprendere servizi come database esterni, API di terze parti, percorsi di connettività di rete verso altri ambienti e servizi DNS. Il primo passo verso un'efficace telemetria delle dipendenze è acquisire una comprensione totale di quali esse siano.
2. Sviluppa una strategia di monitoraggio: una volta acquisito un quadro chiaro delle dipendenze esterne, progetta una strategia di monitoraggio ad hoc per esse. Trovare la strategia giusta implica comprendere le criticità di tutte le dipendenze, il loro comportamento previsto e gli eventuali accordi od obiettivi sul livello di servizio associato (SLA o SLT). Imposta avvisi proattivi che ti informino riguardo a cambiamenti di stato o deviazioni delle prestazioni.
3. Usa il monitoraggio della rete: utilizza il monitoraggio di Internet e il monitoraggio della rete che forniscono approfondimenti completi sulle condizioni globali di Internet e della rete. Questi strumenti consentono di comprendere e rispondere alle interruzioni, ai malfunzionamenti o al degrado delle prestazioni che influiscono sulle dipendenze esterne.
4. Non perdere alcun aggiornamento con AWS Health Dashboard: fornisce avvisi e indicazioni per la correzione qualora AWS sia interessato da eventi che potrebbero influire sui servizi.
    a. Monitora gli eventi AWS Health con le regole Amazon EventBridge o integra a livello di programmazione l'API AWS Health per automatizzare le azioni quando ricevi eventi AWS Health. Possono essere azioni generali, come l'invio di tutti i messaggi pianificati sugli eventi del ciclo di vita a un'interfaccia di chat, oppure azioni specifiche, come l'avvio di un flusso di lavoro in uno strumento di gestione dei servizi IT.
    b. Se usi AWS Organizations, aggrega gli eventi AWS Health tra gli account.
5. Strumenta l'applicazione con AWS X-Ray: AWS X-Ray fornisce approfondimenti sulle prestazioni delle applicazioni e delle relative dipendenze sottostanti. La tracciatura delle richieste dall'inizio alla fine ti permette di identificare colli di bottiglia o guasti nei servizi o nei componenti esterni su cui si basa l'applicazione.
6. Usa Amazon DevOps Guru: questo servizio basato sul machine learning identifica i problemi operativi, prevede quando potrebbero verificarsi problemi critici e consiglia azioni specifiche da intraprendere. Fornisce un supporto prezioso per acquisire approfondimenti sulle dipendenze e assicurarsi che queste non siano la fonte di problemi operativi.
7. Monitora regolarmente: monitora le metriche e i log relativi alle dipendenze esterne in maniera costante. Imposta avvisi per comportamenti imprevisti o prestazioni ridotte.

8. Convalida dopo le modifiche: ogni volta che una dipendenza esterna è interessata da un aggiornamento o una modifica, convalidane le prestazioni e verifica che queste siano in linea con i requisiti dell'applicazione.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS04-BP01 Definizione dei KPI del carico di lavoro
- OPS04-BP02 Implementazione della telemetria dell'applicazione
- OPS04-BP03 Implementazione della telemetria dell'attività degli utenti
- OPS04-BP05 Implementazione della tracciabilità delle transazioni
- OP08-BP04 Creare avvisi fruibili

Documenti correlati:

- Amazon Personal AWS Health Dashboard User Guide
- AWS Internet Monitor User Guide
- AWS X-Ray Developer Guide
- AWS DevOps Guru User Guide

Video correlati:

- Visibility into how internet issues impact app performance
- Introduction to Amazon DevOps Guru
- Manage resource lifecycle events at scale with AWS Health

Esempi correlati:

- Gaining operational insights with AIOps using Amazon DevOps Guru
- AWS Health Aware
- Using Tag-Based Filtering to Manage AWS Health Monitoring and Alerting at Scale

OPS04-BP05 Implementazione del tracciamento distribuito

Il tracciamento distribuito offre un modo per monitorare e visualizzare le richieste mentre attraversano vari componenti di un sistema distribuito. Acquisendo i dati di tracciamento da più fonti e analizzandoli in una vista unificata, i team possono comprendere meglio il flusso delle richieste, in quali punti sono presenti colli di bottiglia e dove devono concentrare gli sforzi di ottimizzazione.

Risultato desiderato: Una visione olistica del flusso delle richieste nel tuo sistema distribuito, che ti permette di ottenere un debug preciso, prestazioni ottimizzate e migliori esperienze utente.

Anti-pattern comuni:

- Strumentazione incoerente: non tutti i servizi in un sistema distribuito sono dotati di strumentazione per il monitoraggio.
- Ignorare la latenza: concentrarsi solo sugli errori e non considerare la latenza o il graduale deterioramento delle prestazioni.

Vantaggi dell'adozione di questa best practice:

- Panoramica completa del sistema: visualizzazione dell'intero percorso delle richieste, dall'ingresso all'uscita.
- Debug avanzato: identificazione rapida dei punti in cui si verificano guasti o problemi di prestazioni.
- Esperienza utente migliorata: monitoraggio e ottimizzazione in base ai dati effettivi dell'utente, garantendo che il sistema soddisfi le esigenze del mondo reale.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Inizia identificando tutti gli elementi del carico di lavoro che richiedono strumentazione. Una volta presi in considerazione tutti i componenti, sfrutta strumenti come AWS X-Ray e OpenTelemetry per raccogliere dati di tracciamento da analizzare con strumenti come X-Ray e Amazon CloudWatchServiceLens Map. Effettua revisioni periodiche insieme agli sviluppatori e integra queste discussioni con strumenti come Amazon DevOps Guru, X-Ray Analytics e X-Ray Insights per ottenere risultati più approfonditi. Imposta avvisi basati sui dati di tracciamento per notificare quando i risultati sono a rischio, come definito nel piano di monitoraggio del carico di lavoro.

Passaggi dell'implementazione

Per implementare il tracciamento distribuito in modo efficace:

1. Adotta AWS X-Ray: implementa X-Ray nella tua applicazione per ottenere informazioni dettagliate sul suo comportamento, comprenderne le prestazioni e individuare i punti critici. Utilizza X-Ray Insights per l'analisi automatica dei tracciamenti.
2. Dota i tuoi servizi di strumenti: verifica che tutti i servizi, dalle funzioni AWS Lambda alle istanze EC2, siano in grado di inviare dati di tracciamento. Più servizi doti di strumentazione, più chiara sarà la visione end-to-end.
3. Incorpora il monitoraggio dell'utente reale tramite CloudWatch e il monitoraggio sintetico: integra il monitoraggio dell'utente reale (RUM) e il monitoraggio sintetico con X-Ray. Ciò ti consente di acquisire esperienze utenti del mondo reale e simulare le interazioni degli utenti per identificare potenziali problemi.
4. Utilizza l'agente CloudWatch: l'agente può inviare dati di tracciamento da X-Ray o da OpenTelemetry, permettendoti di raccogliere informazioni più approfondite.
5. Utilizza Amazon DevOps Guru: DevOps Guru utilizza dati provenienti da X-Ray, CloudWatch, AWS Config e AWS CloudTrail per fornire suggerimenti fruibili.
6. Analizza le tracce: esamina regolarmente i dati di tracciamento per individuare schemi, anomalie o colli di bottiglia che possono influire sulle prestazioni dell'applicazione.
7. Imposta avvisi: configura avvisi in CloudWatch per segnalare schemi insoliti o latenze prolungate, il che ti permette di effettuare una risoluzione proattiva dei problemi.
8. Miglioramento continuo: riesamina la tua strategia di tracciamento man mano che aggiungi o modifichi servizi per acquisire tutti i punti dati pertinenti.

Livello di impegno per il piano di implementazione: Medio

Risorse

Best practice correlate:

- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS04-BP02 Implementazione della telemetria dell'applicazione
- OPS04-BP03 Implementazione della telemetria dell'esperienza utente
- OPS04-BP04 Implementazione della telemetria delle dipendenze

Documenti correlati:

- Guida per gli sviluppatori AWS X-Ray
- Amazon CloudWatch agent User Guide
- Amazon DevOps Guru User Guide

Video correlati:

- Use AWS X-Ray Insights
- AWS on Air ft. Observability: Amazon CloudWatch and AWS X-Ray

Esempi correlati:

- Instrumenting your Application with AWS X-Ray

OPS 5. In che modo riduci i difetti, favorisci la correzione e migliori il flusso nella produzione?

Adotta prassi che migliorino il flusso delle modifiche nella produzione, che attivino la rifattorizzazione e il feedback veloce su qualità e correzione di errori. Tali prassi accelerano l'ingresso in produzione delle modifiche vantaggiose, contengono i problemi che si sono diffusi e permettono di ottenere una rapida identificazione e risoluzione dei problemi introdotti attraverso le attività di implementazione.

Best practice

- OPS05-BP01 Utilizzo del controllo delle versioni
- OPS05-BP02 Test e convalida delle modifiche
- OPS05-BP03 Utilizzo di sistemi di gestione delle configurazioni
- OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione
- OPS05-BP05 Esecuzione della gestione delle patch
- OPS05-BP06 Condivisione degli standard di progettazione
- OPS05-BP07 Implementazione di prassi per migliorare la qualità del codice
- OPS05-BP08 Utilizzo di più ambienti
- OPS05-BP09 Applicazione di modifiche frequenti, minime e reversibili
- OPS05-BP10 Automazione completa dell'integrazione e della distribuzione

OPS05-BP01 Utilizzo del controllo delle versioni

Utilizza il controllo delle versioni per attivare il monitoraggio di modifiche e rilasci.

Molti servizi AWS offrono funzionalità di controllo delle versioni. Utilizza una revisione o un sistema di controllo del codice sorgente come AWS CodeCommit per gestire il codice e altri artefatti, come i modelli AWS CloudFormation controllati dalla versione della tua infrastruttura.

Risultato desiderato: I tuoi team collaborano alla gestione del codice. Una volta unito, il codice è coerente e nessuna modifica viene persa. Gli errori possono essere facilmente ripristinati mediante il corretto controllo delle versioni.

Anti-pattern comuni:

- Hai sviluppato e archiviato il codice sulla workstation. Si è verificato un errore di archiviazione non recuperabile sulla workstation e il codice è andato perso.
- Dopo aver sovrascritto il codice esistente con le modifiche, riavvii l'applicazione e non è più utilizzabile. Non è possibile ripristinare la modifica.
- Hai un blocco di scrittura su un file di report che deve essere modificato da altri utenti. Ti contattano per chiederti di smettere di utilizzarlo in modo che possano completare le loro attività.
- Il team di ricerca ha lavorato a un'analisi dettagliata che definisce il tuo lavoro futuro. Qualcuno ha salvato accidentalmente la lista della spesa nel report finale. Non puoi ripristinare la modifica e devi ricreare il report.

Vantaggi dell'adozione di questa best practice: Grazie alle funzionalità di controllo delle versioni, puoi ripristinare facilmente gli stati validi noti e le versioni precedenti e limitare il rischio di perdita degli asset.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Mantieni gli asset in repository con controllo delle versioni. In questo modo si supporta il monitoraggio delle modifiche, la distribuzione di nuove versioni, il rilevamento delle modifiche apportate alle versioni esistenti e il ripristino delle versioni precedenti, ad esempio il rollback a uno stato corretto noto in caso di errore. Integra nelle tue procedure le funzionalità di controllo delle versioni dei sistemi di gestione delle configurazioni.

Risorse

Best practice correlate:

- OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione

Documenti correlati:

- Che cos'è AWS CodeCommit?

Video correlati:

- Introduzione ad AWS CodeCommit

OPS05-BP02 Test e convalida delle modifiche

Ogni modifica apportata deve essere testata per evitare errori in produzione. Questa best practice si concentra sulla verifica delle modifiche dal controllo di versione alla creazione dell'artefatto. Oltre alle modifiche al codice dell'applicazione, i test dovrebbero includere l'infrastruttura, la configurazione, i controlli di sicurezza e le procedure operative. I test assumono molte forme, dai test unitari all'analisi dei componenti software (SCA). Spostando i test più a sinistra nel processo di integrazione e consegna del software ottieni una maggiore certezza della qualità degli artefatti.

L'organizzazione deve sviluppare standard di test per tutti gli artefatti software. I test automatizzati riducono la fatica ed evitano gli errori dei test manuali. I test manuali potrebbero essere necessari in alcuni casi. Gli sviluppatori devono avere accesso ai risultati dei test automatizzati per creare cicli di feedback che migliorino la qualità del software.

Risultato desiderato: le modifiche software vengono testate prima del rilascio. Gli sviluppatori hanno accesso ai risultati dei test e alle convalide. La tua organizzazione ha uno standard per i test che applica a tutte le modifiche software.

Anti-pattern comuni:

- Implementi una nuova modifica software senza test. Non funziona in produzione e genera un'interruzione.
- I nuovi gruppi di sicurezza vengono implementati con AWS CloudFormation senza essere testati in un ambiente di pre-produzione. I gruppi di sicurezza rendono la tua app irraggiungibile per i clienti.

- Un metodo viene modificato, ma non ci sono test di unità. Il software ha esito negativo quando viene implementato in produzione.

Vantaggi dell'adozione di questa best practice: si riduce la percentuale di modifiche non riuscite nelle implementazioni del software. La qualità del software viene migliorata. Gli sviluppatori hanno una maggiore consapevolezza della fattibilità del loro codice. Le policy di sicurezza possono essere implementate in maniera affidabile per supportare la conformità dell'organizzazione. Le modifiche all'infrastruttura, come gli aggiornamenti automatici delle politiche di scaling, vengono testate in anticipo per soddisfare le esigenze del traffico.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

I test vengono eseguiti su tutte le modifiche, dal codice dell'applicazione all'infrastruttura, come parte della pratica di integrazione continua. I risultati dei test vengono pubblicati in modo che gli sviluppatori abbiano un feedback rapido. La tua organizzazione ha uno standard per i test che applica a tutte le modifiche software.

Usa la potenza dell'IA generativa con Amazon Q Developer per migliorare la produttività degli sviluppatori e la qualità del codice. Amazon Q Developer include la generazione di suggerimenti di codice (basati su modelli linguistici di grandi dimensioni), la produzione di unit test (comprese le condizioni limite) e il miglioramento della sicurezza del codice tramite il rilevamento e la correzione delle vulnerabilità di sicurezza.

Esempio del cliente

Nell'ambito della pipeline di integrazione continua, AnyCompany Retail esegue diversi tipi di test su tutti gli artefatti software. Praticano lo sviluppo guidato dai test, per cui tutto il software è dotato di test unitari. Una volta creato l'artefatto, eseguono test end-to-end. Al termine di questa prima serie di test, viene eseguita una scansione statica della sicurezza dell'applicazione, alla ricerca di vulnerabilità note. Gli sviluppatori ricevono messaggi al superamento di ciascun gate di test. Una volta completati tutti i test, l'artefatto software viene archiviato in un repository di artefatti.

Passaggi dell'implementazione

1. Collaborare con le parti interessate dell'organizzazione per sviluppare uno standard di test per gli artefatti software. Quali test standard devono superare tutti gli artefatti? Ci sono requisiti di conformità o di governance che devono essere inclusi nella copertura dei test? Devi condurre test di qualità del codice? Quando i test sono terminati, chi deve esserne a conoscenza?

1. L'architettura di riferimento della pipeline di distribuzione AWS contiene un elenco autorevole dei tipi di test che possono essere condotti su artefatti software come parte di una pipeline di integrazione.

2. Fornisci la tua applicazione dei test necessari in base allo standard di test del software. Ogni set di test deve essere completato in meno di dieci minuti. I test devono essere eseguiti come parte della pipeline di integrazione.
   a. Usa Amazon Q Developer, uno strumento di IA generativa che consente di creare casi di unit test (comprese le condizioni limite), generare funzioni utilizzando codice e commenti e implementare gli algoritmi noti.
   b. Usa Amazon CodeGuru Reviewer per testare il codice dell'applicazione e trovare eventuali difetti.
   c. Puoi usare AWS CodeBuild per condurre i test su artefatti software.
   d. AWS CodePipeline può organizzare i tuoi test software in una pipeline.

Risorse

Best practice correlate:

- OPS05-BP01 Utilizzo del controllo delle versioni
- OPS05-BP06 Condivisione degli standard di progettazione
- OPS05-BP07 Implementazione di prassi per migliorare la qualità del codice
- OPS05-BP10 Automazione completa dell'integrazione e dell'implementazione

Documenti correlati:

- Adopt a test-driven development approach
- Accelerate your Software Development Lifecycle with Amazon Q
- Amazon Q Developer, now generally available, includes previews of new capabilities to reimagine developer experience
- The Ultimate Cheat Sheet for Using Amazon Q Developer in Your IDE
- Shift-Left Workload, leveraging AI for Test Creation
- Amazon Q Developer Center
- 10 ways to build applications faster with Amazon CodeWhisperer
- Looking beyond code coverage with Amazon CodeWhisperer

- Best Practices for Prompt Engineering with Amazon CodeWhisperer
- Automated AWS CloudFormation Testing Pipeline with TaskCat and CodePipeline
- Building end-to-end AWS DevSecOps CI/CD pipeline with open source SCA, SAST, and DAST tools
- Getting started with testing serverless applications
- My CI/CD pipeline is my release captain
- Practicing Continuous Integration and Continuous Delivery on AWS Whitepaper

Video correlati:

- Implement an API with Amazon Q Developer Agent for Software Development
- Installing, Configuring, & Using Amazon Q Developer with JetBrains IDEs (How-to)
- Mastering the art of Amazon CodeWhisperer - YouTube playlist
- AWS re:Invent 2020: Testable infrastructure: Integration testing on AWS
- AWS Summit ANZ 2021 - Driving a test-first strategy with CDK and test driven development
- Testing Your Infrastructure as Code with AWS CDK

Risorse correlate:

- Building applications using generative AI with Amazon CodeWhisperer
- Amazon CodeWhisperer Workshop
- AWS Deployment Pipeline Reference Architecture - Application
- AWS Kubernetes DevSecOps Pipeline
- Policy as Code Workshop – Test Driven Development
- Run unit tests for a Node.js application from GitHub by using AWS CodeBuild
- Use Serverspec for test-driven development of infrastructure code

Servizi correlati:

- Amazon Q Developer
- Amazon CodeGuru Reviewer
- AWS CodeBuild
- AWS CodePipeline

OPS05-BP03 Utilizzo di sistemi di gestione delle configurazioni

L'utilizzo di sistemi di gestione delle configurazioni permette di effettuare modifiche alle stesse e tenerne traccia. Questi sistemi riducono gli errori causati dai processi manuali e il livello di impegno richiesto per la distribuzione delle modifiche.

Durante l'inizializzazione di una risorsa, la gestione delle configurazioni statiche consente di impostare valori che dovrebbero rimanere coerenti per tutta la vita utile della risorsa. Ne sono alcuni esempi l'azione di configurare un server web o applicativo su un'istanza oppure di definire la configurazione di un servizio AWS nella AWS Management Console o tramite la AWS CLI.

Al momento dell'inizializzazione, la gestione delle configurazioni dinamiche consente di impostare valori che possono cambiare nel corso della vita utile di una risorsa. Ad esempio è possibile impostare un interruttore funzionale in grado di attivare una funzionalità nel codice tramite una modifica della configurazione, oppure modificare il livello di dettaglio del log durante un incidente per acquisire un maggior numero di dati e cambiarlo in seguito per tornare al livello di dettaglio precedente, risparmiando così in numero di log e nei relativi costi.

In AWS, è possibile utilizzare AWS Config per monitorare in modo continuo le configurazioni delle risorse AWS tra i diversi account e regioni. Questa soluzione aiuta a tenere traccia della cronologia delle configurazioni, a capire che effetto avrebbe la modifica di una configurazione sulle altre risorse e a verificarle rispetto alle configurazioni previste o desiderate tramite Regole di AWS Config e pacchetti di conformità AWS Config.

Se sulle applicazioni in esecuzione su istanze Amazon EC2, AWS Lambda, container, funzioni serverless, applicazioni mobili o dispositivi IoT sono attive configurazioni dinamiche, è possibile utilizzare AWS AppConfig per configurarle, convalidarle, implementarle e monitorarle nei tuoi ambienti.

In AWS, puoi creare pipeline di integrazione continua/distribuzione continua (CI/CD) utilizzando servizi come gli Strumenti per sviluppatori in AWS (ad esempio: AWS CodeCommit, AWS CodeBuild, AWS CodePipeline, AWS CodeDeploye AWS CodeStar).

Risultato desiderato: Puoi configurare, convalidare e implementare come parte della tua pipeline di integrazione continua e di distribuzione continua (CI/CD). Esegui il monitoraggio per verificare che le configurazioni siano corrette. Ciò riduce al minimo l'impatto sugli utenti finali e sui clienti.

Anti-pattern comuni:

- Aggiorni manualmente la configurazione del server Web all'interno del parco istanze e un certo numero di server non risponde a causa di errori di aggiornamento.

- Aggiorni manualmente il parco istanze del server applicazioni nel corso di molte ore. L'incoerenza nella configurazione durante la modifica causa comportamenti imprevisti.
- Qualcuno ha aggiornato i tuoi gruppi di sicurezza e i server Web non sono più accessibili. Senza sapere cosa è stato modificato, dedichi molto tempo a esaminare il problema prolungando il tempo necessario per il ripristino.
- Avvii una configurazione di preproduzione in produzione tramite CI/CD senza una convalida. Esponi utenti e clienti a dati e servizi errati.

Vantaggi dell'adozione di questa best practice: L'adozione di sistemi di gestione della configurazione riduce il livello di impegno necessario per apportare e tenere traccia delle modifiche e la frequenza degli errori causati dalle procedure manuali. I sistemi di gestione della configurazione forniscono garanzie per quanto riguarda la governance, la conformità e i requisiti normativi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I sistemi di gestione della configurazione vengono utilizzati per tenere traccia e implementare le modifiche nelle configurazioni delle applicazioni e degli ambienti. I sistemi di gestione della configurazione vengono utilizzati anche per ridurre gli errori causati dai processi manuali, rendere le modifiche alla configurazione ripetibili e verificabili e per ridurre il livello di impegno.

Passaggi dell'implementazione

1. Identifica i proprietari della configurazione.
   a. Metti a conoscenza i proprietari delle configurazioni di qualsiasi esigenza di conformità, governance o normativa.
2. Identifica gli elementi e i risultati della configurazione.
   a. Gli elementi di configurazione sono tutte le configurazioni ambientali e dell'applicazione interessate da un'implementazione all'interno della pipeline CI/CD.
   b. I risultati finali includono criteri di successo, convalide e aspetti da monitorare.
3. Seleziona gli strumenti per la gestione della configurazione in base ai requisiti aziendali e alla pipeline di distribuzione.
4. Per modifiche significative alla configurazione, prendi in considerazione le implementazioni ponderate, ad esempio le implementazioni canary, per ridurre al minimo l'impatto di configurazioni errate.
5. Integra la gestione della configurazione nella tua pipeline CI/CD.

6. Convalida tutte le modifiche inserite.

Risorse

Best practice correlate:

- OPS06-BP01 Preparazione di un piano in caso di esito negativo delle modifiche
- OPS06-BP02 Implementazioni dei test
- OPS06-BP03 Utilizza strategie di deployment sicure
- OPS06-BP04 Automazione dei test e del rollback

Documenti correlati:

- AWS Control Tower
- AWS Landing Zone Accelerator
- AWS Config
- What is AWS Config?
- AWS AppConfig
- What is AWS CloudFormation?
- Strumenti per sviluppatori in AWS

Video correlati:

- AWS re:Invent 2022 - Proactive governance and compliance for AWS workloads
- AWS re:Invent 2020: Achieve compliance as code using AWS Config
- Manage and Deploy Application Configurations with AWS AppConfig

OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione

Utilizza sistemi di gestione della creazione e distribuzione Questi sistemi riducono gli errori causati dai processi manuali e il livello di impegno richiesto per la distribuzione delle modifiche.

In AWS, puoi compilare pipeline di integrazione continua/implementazione continua (CI/CD) utilizzando servizi come gli Strumenti per sviluppatori in AWS (ad esempio, AWS CodeCommit, AWS CodeBuild, AWS CodePipeline, AWS CodeDeploye AWS CodeStar).

Risultato desiderato: I sistemi di gestione della costruzione e dell'implementazione supportano il sistema di distribuzione e integrazione continua (CI/CD) dell'organizzazione, che fornisce funzionalità per automatizzare rollout sicuri con le configurazioni corrette.

Anti-pattern comuni:

- Dopo aver compilato il codice nel sistema di sviluppo, copi il file eseguibile nei sistemi di produzione e questo non si avvia. I file di log locali indicano che l'operazione è risultata impossibile a causa della mancanza di dipendenze.
- Hai creato l'applicazione con nuove funzionalità nel tuo ambiente di sviluppo e fornisci il codice per eseguire il controllo qualità (QA). Il controllo qualità non riesce perché mancano asset statici.
- Venerdì, dopo un notevole sforzo, hai creato l'applicazione manualmente nel tuo ambiente di sviluppo, incluse le nuove funzionalità codificate. Lunedì non sei in grado di ripetere le fasi che ti hanno consentito di creare correttamente la tua applicazione.
- Esegui i test creati per la nuova versione. Quindi passi la settimana successiva a configurare un ambiente di test ed eseguire tutti i test di integrazione esistenti seguiti dai test delle prestazioni. Il nuovo codice ha un impatto inaccettabile sulle prestazioni e deve essere risviluppato e quindi ritestato.

Vantaggi dell'adozione di questa best practice: Fornendo meccanismi per gestire le attività di compilazione e distribuzione, riduci il livello di impegno necessario per eseguire attività ripetitive, consenti ai membri del team di concentrarsi liberamente sulle loro attività creative di valore elevato e limiti l'introduzione di errori derivanti da procedure manuali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I sistemi di gestione della creazione e implementazione vengono utilizzati per tenere traccia e implementare le modifiche, ridurre gli errori causati dai processi manuali e diminuire il livello di impegno richiesto per le implementazioni sicure. Automatizza completamente la pipeline di integrazione e distribuzione dal check-in del codice fino alle fasi di creazione, test, distribuzione e convalida. Ciò riduce il lead time e i costi, incoraggia una maggiore frequenza delle modifiche, riduce il livello di impegno e aumenta la collaborazione.

Passaggi dell'implementazione

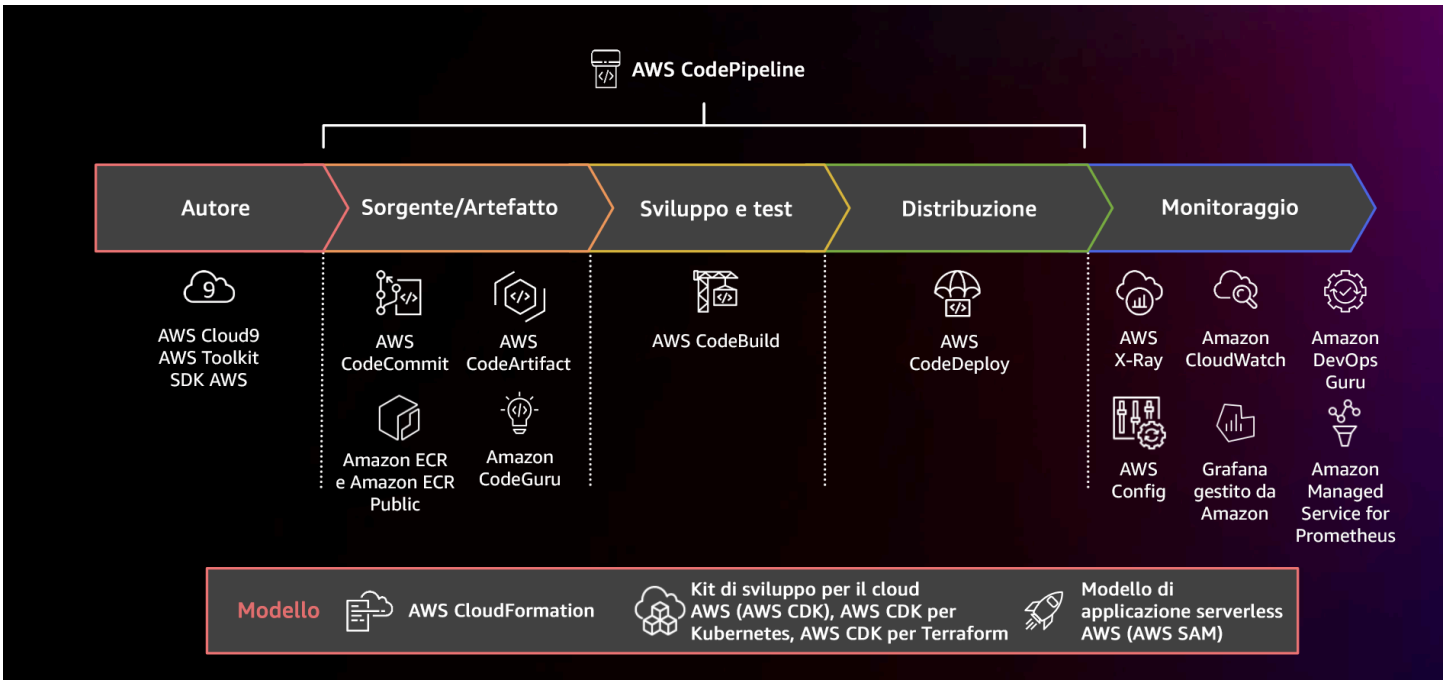


Diagramma che mostra una pipeline CI/CD che utilizza AWS CodePipeline e servizi correlati

1. Utilizza AWS CodeCommit per verificare la versione, archiviare e gestire risorse come documenti, codice sorgente e file binari.
2. Utilizza CodeBuild per compilare il codice sorgente, eseguire test delle unità e produrre artefatti pronti per l'implementazione.
3. Utilizza CodeDeploy come servizio di implementazione per automatizzare l'implementazione delle applicazioni su istanze Amazon EC2 , istanze on-premise, funzioni AWS Lambda serverlesso Amazon ECS.
4. Monitora le tue implementazioni.

Risorse

Best practice correlate:

- OPS06-BP04 Automazione dei test e del rollback

Documenti correlati:

- Strumenti per sviluppatori in AWS

- Che cos'è AWS CodeCommit?
- Che cos'è AWS CodeBuild?
- AWS CodeBuild
- Che cos'è AWS CodeDeploy?

Video correlati:

- AWS re:Invent 2022 - AWS Well-Architected best practices for DevOps on AWS

OPS05-BP05 Esecuzione della gestione delle patch

La gestione delle patch consente di ottenere funzionalità, risolvere problemi e rispettare i requisiti di governance. Automatizza la gestione delle patch per ridurre gli errori causati dai processi manuali, dimensionare e ridurre il livello di impegno richiesto per applicare le patch.

La gestione delle patch e delle vulnerabilità fa parte delle attività di gestione dei rischi e dei vantaggi. È preferibile disporre di infrastrutture immutabili e distribuire carichi di lavoro in stati noti verificati. Se ciò non è realizzabile, l'applicazione di patch sul posto è l'alternativa.

Amazon EC2 Image Builder fornisce pipeline per l'aggiornamento di immagini AMI. Come parte della gestione delle patch, prendi in considerazione l'utilizzo di Amazon Machine Image (AMI) con una pipeline di immagini AMI o immagini del container con una pipeline di immagini Docker. Inoltre, puoi utilizzare AWS Lambda, che fornisce modelli per runtime personalizzati e librerie aggiuntive per eliminare le vulnerabilità.

È consigliabile gestire gli aggiornamenti alle Amazon Machine Image per immagini Linux o Windows Server utilizzando Amazon EC2 Image Builder. Puoi utilizzare Amazon Elastic Container Registry (Amazon ECR) con la pipeline esistente per gestire le immagini Amazon ECS ed Amazon EKS. Lambda include funzionalità di gestione delle versioni.

L'applicazione di patch non deve essere eseguita sui sistemi di produzione senza prima eseguire test in un ambiente sicuro. Le patch devono essere applicate solo se supportano risultati operativi o aziendali. In AWS, è possibile utilizzare Gestione patch di AWS Systems Manager per automatizzare il processo di applicazione di patch ai sistemi gestiti e pianificare l'attività utilizzando le finestre di manutenzione di Systems Manager.

Risultato desiderato: Le immagini AMI e dei container sono aggiornate, dotate di patch e pronte per il lancio. È possibile tenere traccia dello stato di tutte le immagini implementate e conoscere

la conformità delle patch. Puoi eseguire report sullo stato attuale e disporre di un processo per soddisfare le tue esigenze di conformità.

Anti-pattern comuni:

- Ti viene assegnato il compito di applicare tutte le nuove patch di sicurezza entro 2 ore, il che provoca più interruzioni a causa dell'incompatibilità dell'applicazione con le patch.
- Una libreria senza patch comporta conseguenze indesiderate in quanto parti sconosciute utilizzano vulnerabilità al suo interno per accedere al carico di lavoro.
- L'applicazione di patch agli ambienti per sviluppatori viene eseguita automaticamente senza avvisare gli sviluppatori. Gli sviluppatori ti inviano più reclami perché il loro ambiente non funziona come previsto.
- Non hai applicato patch al software pronto all'uso commerciale su un'istanza persistente. Quando hai problemi con il software e contatti il fornitore, questo ti informa che la versione non è supportata e che devi applicare le patch a un livello specifico per ricevere assistenza.
- Una patch rilasciata di recente per il software di crittografia utilizzato offre miglioramenti significativi in termini di prestazioni. Il sistema privo di patch presenta problemi di prestazioni che rimangono in vigore a causa della mancata applicazione di patch.
- Ricevi una notifica di una vulnerabilità zero-day che richiede una correzione di emergenza; quindi devi applicare manualmente le patch a tutti i tuoi ambienti.

Vantaggi dell'adozione di questa best practice: Stabilendo un processo di gestione delle patch, inclusi i criteri per l'applicazione di patch e la metodologia di distribuzione tra gli ambienti, sarai in grado di dimensionare e generare report sui livelli di patch. Ciò fornisce garanzie sull'applicazione delle patch di sicurezza e una chiara visibilità sullo stato delle correzioni note in atto. Ciò incoraggia l'adozione delle caratteristiche e funzionalità desiderate, aiuta a eliminare rapidamente i problemi e a mantenere la conformità alla governance. Implementa sistemi di gestione delle patch e automazione per ridurre il livello di impegno per distribuire le patch e limitare gli errori causati dai processi manuali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Applica patch ai sistemi per correggere gli errori, ottenere le funzionalità o le capacità desiderate e assicurare la conformità alle policy di governance e ai requisiti di supporto del vendor. Nei sistemi immutabili, distribuisci con il set di patch appropriato per raggiungere il risultato desiderato. Automatizza il meccanismo di gestione delle patch per ridurre il tempo necessario per applicare le

patch, evitare gli errori causati dai processi manuali e diminuire il livello di impegno richiesto per applicare le patch.

Passaggi dell'implementazione

Per Amazon EC2 Image Builder:

1. specifica i dettagli della pipeline utilizzando Amazon EC2 Image Builder:
   a. Crea una pipeline di immagini e assegnale un nome
   b. Definisci la pianificazione e il fuso orario della pipeline
   c. Configura eventuali dipendenze
2. Scegli una ricetta:
   a. Seleziona una ricetta esistente o creane una nuova
   b. Seleziona il tipo di immagine
   c. Assegna un nome e una versione alla tua ricetta
   d. Seleziona l'immagine di base
   e. Aggiungi componenti di compilazione e inseriscili nel registro di destinazione
3. Facoltativo: definisci la configurazione dell'infrastruttura.
4. Facoltativo: definisci le impostazioni di configurazione.
5. Revisiona le impostazioni.
6. Mantieni il livello di igiene delle ricette a livelli ottimali.

Per Gestione patch di Systems Manager:

1. Crea una patch di base.
2. Seleziona un metodo per le operazioni di definizione del percorso.
3. Abilita il report e la scansione della conformità.

Risorse

Best practice correlate:

- OPS06-BP04 Automazione dei test e del rollback

Documenti correlati:

- What is Amazon EC2 Image Builder
- Create an image pipeline using the Amazon EC2 Image Builder
- Create a container image pipeline
- Gestione patch di AWS Systems Manager
- Working with Patch Manager
- Working with patch compliance reports
- Strumenti per sviluppatori in AWS

Video correlati:

- CI/CD per applicazioni serverless su AWS
- Progettare nell'ottica Ops

Esempi correlati:

- Well-Architected Labs: Inventario e gestione delle patch
- AWS Systems Manager Patch Manager tutorials

OPS05-BP06 Condivisione degli standard di progettazione

Condividi le best practice con i team per incrementare la consapevolezza e potenziare al massimo i vantaggi delle attività di sviluppo. Documentale e mantienile aggiornate di pari passo con l'evoluzione dell'architettura. Se nella tua organizzazione vengono applicati standard condivisi, è fondamentale che esistano meccanismi per richiedere aggiunte, modifiche ed eccezioni agli standard. Senza questa opzione, gli standard diventano un ostacolo per l'innovazione.

Risultato desiderato: Gli standard di progettazione vengono condivisi fra team nelle organizzazioni. Vengono documentati e tenuti aggiornati in base all'evoluzione delle best practice.

Anti-pattern comuni:

- Due team di sviluppo hanno creato ciascuno un servizio di autenticazione utente. Gli utenti devono mantenere un set separato di credenziali per ogni parte del sistema a cui vogliono accedere.
- Ogni team gestisce la propria infrastruttura. Un nuovo requisito di conformità impone una modifica all'infrastruttura e ogni team la implementa in modo diverso.

Vantaggi dell'adozione di questa best practice: L'uso di standard condivisi incoraggia l'applicazione di best practice e permette di ottenere i massimi vantaggi dalle attività di sviluppo. La documentazione e l'aggiornamento degli standard di progettazione tengono l'organizzazione al passo con le best practice e i requisiti di sicurezza e conformità.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Condividi le best practice, gli standard di progettazione, gli elenchi di controllo, le procedure operative, le linee guida e i requisiti di governance esistenti tra team diversi. Definisci procedure per richiedere modifiche, aggiunte ed eccezioni agli standard di progettazione per supportare il miglioramento e l'innovazione. Rendi noto ai team il contenuto pubblicato. Predisponi un meccanismo per mantenere aggiornati gli standard di progettazione in base all'emergere di nuove best practice.

Esempio del cliente

AnyCompany Retail ha un team interfunzionale che crea modelli di architettura software. Questo team crea l'architettura con conformità e governance integrate. I team che adottano gli standard condivisi traggono vantaggio dall'integrazione di conformità e governance. Possono creare rapidamente soluzioni sulla base degli standard di progettazione. Il team responsabile dell'architettura si incontra ogni trimestre per valutare i modelli architetturali e aggiornarli, se necessario.

Passaggi dell'implementazione

1. Identifica un team interfunzionale responsabile dello sviluppo e dell'aggiornamento degli standard di progettazione. Questo team collaborerà con gli stakeholder in tutta l'organizzazione per sviluppare standard di progettazione, procedure operative, elenchi di controllo, linee guida e requisiti di governance. Documenta gli standard di progettazione e condividili internamente all'organizzazione.
   a. AWS Service Catalog può aiutarti a creare portfolio che rappresentano gli standard di progettazione usando il modello Infrastruttura come codice (IaC). Puoi condividere portfolio tra più account.
2. Predisponi un meccanismo per mantenere aggiornati gli standard di progettazione man mano che vengono identificate nuove best practice.
3. Se gli standard di progettazione vengono applicati a livello centrale, definisci un processo per richiedere modifiche, aggiornamenti ed eccezioni.

Livello di impegno per il piano di implementazione: medio. Lo sviluppo di un processo per creare e condividere standard di progettazione può richiedere il coordinamento e la cooperazione con gli stakeholder in tutta l'organizzazione.

Risorse

Best practice correlate:

- OPS01-BP03 Valutazione dei requisiti di governance – I requisiti di governance influiscono sugli standard di progettazione.
- OPS01-BP04 Valutazione dei requisiti di conformità – La conformità è un fattore essenziale nella creazione di standard di progettazione.
- OPS07-BP02 Revisione costante della prontezza operativa – Gli elenchi di controllo della prontezza operativa sono un meccanismo per implementare standard di progettazione durante la progettazione del carico di lavoro.
- OPS11-BP01 Definizione di un processo per il miglioramento continuo – L'aggiornamento degli standard di progettazione contribuisce a un miglioramento continuo.
- OPS11-BP04 Gestione delle informazioni – Nell'ambito della procedura di gestione delle informazioni, documenta e condividi gli standard di progettazione.

Documenti correlati:

- Automate AWS Backups with AWS Service Catalog
- AWS Service Catalog Account Factory-Enhanced
- Expedia Group crea un'offerta Database as a Service (DBaaS) usando il AWS Service Catalog
- Mantenimento della visibilità sull'uso di modelli architetturali cloud
- Simplify sharing your AWS Service Catalog portfolios in an AWS Organizations setup

Video correlati:

- AWS Service Catalog – Getting Started
- AWS re:Invent 2020: Manage your AWS Service Catalog portfolios like an expert

Esempi correlati:

- AWS Service Catalog Reference Architecture

- AWS Service Catalog Workshop

Servizi correlati:

- AWS Service Catalog

OPS05-BP07 Implementazione di prassi per migliorare la qualità del codice

Implementa prassi per migliorare la qualità del codice e ridurre al minimo i difetti. Alcuni esempi includono sviluppo basato su test, revisioni del codice, adozione degli standard e programmazione in coppia. Inserisci queste prassi nel processo di integrazione continua e distribuzione continua.

Risultato desiderato: l'organizzazione usa best practice come le revisioni del codice e la programmazione in coppia per migliorare la qualità del codice. Sviluppatori e operatori adottano le best practice per la qualità del codice nell'ambito del ciclo di vita di sviluppo del software.

Anti-pattern comuni:

- Commit del codice nel ramo principale dell'applicazione senza alcuna revisione. In questo modo, la modifica viene automaticamente implementata nell'ambiente di produzione e causa un'interruzione.
- Sviluppo di una nuova applicazione senza unit test, test end-to-end o test di integrazione. Non è possibile in alcun modo testare l'applicazione prima dell'implementazione.
- I team apportano modifiche manuali nell'ambiente di produzione per gestire gli errori. Le modifiche non vengono sottoposte a test o revisioni del codice, né vengono acquisite o registrate durante i processi di integrazione continua e distribuzione continua.

Vantaggi dell'adozione di questa best practice: l'adozione delle procedure per migliorare la qualità del codice ti consente di ridurre al minimo i problemi di produzione. La qualità del codice facilita l'uso delle best practice, come la programmazione in coppia, le revisioni del codice e l'implementazione di strumenti di produttività basati sull'IA.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Implementa prassi per migliorare la qualità del codice in modo da ridurre gli errori prima dell'implementazione. Usa prassi come lo sviluppo basato su test, le revisioni del codice e la programmazione in coppia per migliorare la qualità dello sviluppo.

Usa la potenza dell'IA generativa con Amazon Q Developer per migliorare la produttività degli sviluppatori e la qualità del codice. Amazon Q Developer include la generazione di suggerimenti di codice (basati su modelli linguistici di grandi dimensioni), la produzione di unit test (comprese le condizioni limite) e il miglioramento della sicurezza del codice tramite il rilevamento e la correzione delle vulnerabilità di sicurezza.

Esempio del cliente

AnyCompany Retail adotta diverse prassi per migliorare la qualità del codice. L'azienda ha adottato lo sviluppo basato su test come standard per la scrittura di applicazioni. Per alcune nuove funzionalità, gli sviluppatori eseguiranno la programmazione in coppia durante uno sprint. Ogni richiesta pull viene sottoposta a una revisione del codice da parte di uno sviluppatore senior prima di essere integrata e implementata.

Passaggi dell'implementazione

1. Adotta prassi per la qualità del codice come lo sviluppo basato su test, le revisioni del codice e la programmazione in coppia nel processo di integrazione continua e distribuzione continua. Usa queste tecniche per migliorare la qualità del software.
   a. Utilizza Amazon Q Developer, uno strumento di IA generativa che consente di creare casi di unit test (comprese le condizioni limite), generare funzioni utilizzando codice e commenti, implementare gli algoritmi noti, rilevare violazioni delle policy di sicurezza e vulnerabilità nel codice, rilevare segreti, analizzare l'infrastructure as code (IaC), documentare il codice e apprendere più rapidamente librerie di codici di terze parti.
   b. Amazon CodeGuru Reviewer può fornire suggerimenti di programmazione per il codice Java e Python tramite il machine learning.
   c. Puoi creare ambienti di sviluppo condivisi con AWS Cloud9 in cui collaborare allo sviluppo del codice.

Livello di impegno per il piano di implementazione: medio. Esistono molti modi per implementare questa best practice, ma la realizzazione dell'adozione da parte dell'organizzazione può essere problematica.

Risorse

Best practice correlate:

- OPS05-BP02 Test e convalida delle modifiche

- OPS05-BP06 Condivisione degli standard di progettazione

Documenti correlati:

- Adopt a test-driven development approach
- Accelerate your Software Development Lifecycle with Amazon Q
- Amazon Q Developer, now generally available, includes previews of new capabilities to reimagine developer experience
- The Ultimate Cheat Sheet for Using Amazon Q Developer in Your IDE
- Shift-Left Workload, leveraging AI for Test Creation
- Amazon Q Developer Center
- 10 ways to build applications faster with Amazon CodeWhisperer
- Looking beyond code coverage with Amazon CodeWhisperer
- Best Practices for Prompt Engineering with Amazon CodeWhisperer
- Agile Software Guide
- My CI/CD pipeline is my release captain
- Automate code reviews with Amazon CodeGuru Reviewer
- Adopt a test-driven development approach
- How DevFactory builds better applications with Amazon CodeGuru
- On Pair Programming
- RENGA Inc. automates code reviews with Amazon CodeGuru
- The Art of Agile Development: Test-Driven Development
- Why code reviews matter (and actually save time!)

Video correlati:

- Implement an API with Amazon Q Developer Agent for Software Development
- Installing, Configuring, & Using Amazon Q Developer with JetBrains IDEs (How-to)
- Mastering the art of Amazon CodeWhisperer - YouTube playlist
- AWS re:Invent 2020: Continuous improvement of code quality with Amazon CodeGuru
- AWS Summit ANZ 2021 - Driving a test-first strategy with CDK and test driven development

Servizi correlati:

- Amazon Q Developer
- Amazon CodeGuru Reviewer
- Amazon CodeGuru Profiler
- AWS Cloud9

OPS05-BP08 Utilizzo di più ambienti

Utilizza ambienti multipli per sperimentare, sviluppare e testare il carico di lavoro. Applica livelli crescenti di controlli man mano che gli ambienti si avvicinano alla fase di produzione per avere la certezza che il carico di lavoro funzioni come previsto una volta implementato.

Risultato desiderato: Disponi di più ambienti che riflettono le tue esigenze di conformità e governance. Testi e promuovi il codice negli ambienti lungo il tuo percorso verso la produzione.

Anti-pattern comuni:

- Stai sviluppando in un ambiente di sviluppo condiviso e un altro sviluppatore sovrascrive le tue modifiche al codice.
- I controlli di sicurezza restrittivi nell'ambiente di sviluppo condiviso impediscono di sperimentare nuovi servizi e funzionalità.
- Esegui test di carico sui tuoi sistemi di produzione e causa un'interruzione per i tuoi utenti.
- Si è verificato un errore critico che ha causato la perdita di dati nella produzione. Nel tuo ambiente di produzione tenti di ricreare le condizioni che portano alla perdita di dati in modo da poter identificare come si è verificata e impedire che si ripeta. Per evitare un'ulteriore perdita di dati durante il test, devi rendere l'applicazione non disponibile per i tuoi utenti.
- Stai operando un servizio multi-tenant e non sei in grado di supportare la richiesta di un cliente per un ambiente dedicato.
- Ogni volta che esegui un test, lo fai nel tuo ambiente di produzione.
- Ritieni che la semplicità di un singolo ambiente prevalga sulla portata dell'impatto che possono avere modifiche all'interno dell'ambiente.

Vantaggi dell'adozione di questa best practice: puoi supportare più ambienti di sviluppo, test e produzione simultanei senza creare conflitti tra sviluppatori o community di utenti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Utilizza più ambienti e fornisci agli sviluppatori ambienti di sperimentazione (sandbox) con controlli minimi per incoraggiare la sperimentazione. Fornisci ambienti di sviluppo individuali per facilitare il lavoro in parallelo, incrementando l'agilità dello sviluppo. Implementa controlli più rigorosi negli ambienti che si avvicinano alla produzione per consentire agli sviluppatori di innovare. Utilizza l'approccio Infrastructure-as-Code e sistemi di gestione delle configurazioni per distribuire ambienti configurati in modo coerente con i controlli presenti in produzione per assicurare che i sistemi funzionino nel modo previsto quando vengono distribuiti. Quando gli ambienti non vengono utilizzati, disattivali per evitare costi associati alle risorse inattive, ad esempio i sistemi di sviluppo nelle ore serali e nei fine settimana. Durante i test di carico, è necessario implementare ambienti equivalenti a quelli di produzione per migliorare la validità dei risultati.

Risorse

Documenti correlati:

- Pianificatore di istanze su AWS
- Che cos'è AWS CloudFormation?

OPS05-BP09 Applicazione di modifiche frequenti, minime e reversibili

Le modifiche frequenti, minime e reversibili riducono la portata e l'impatto di una modifica. Le modifiche frequenti, minime e reversibili, se effettuate utilizzando congiuntamente sistemi di gestione delle modifiche, di gestione della configurazione e di compilazione e distribuzione, riducono la portata e l'impatto di una modifica. Questo si traduce in una risoluzione dei problemi più efficace, accelerando la correzione e mantenendo la possibilità di rollback delle modifiche.

Anti-pattern comuni:

- Distribuisci una nuova versione della tua applicazione ogni trimestre con una finestra di modifica, il che comporta la disattivazione di un servizio di base.
- Spesso apporti modifiche allo schema del database senza che ne venga tenuta traccia nei sistemi di gestione.
- Esegui aggiornamenti manuali sul posto, sovrascrivendo le installazioni e le configurazioni esistenti, senza avere un chiaro piano di rollback.

Vantaggi dell'adozione di questa best practice: Le attività di sviluppo sono più rapide grazie all'implementazione frequente di modifiche minime. Quando le modifiche sono minime, è molto più semplice identificare se hanno conseguenze indesiderate e, in tal caso, ripristinare la condizione precedente. Quando le modifiche sono reversibili, il rischio di implementare le modifiche è minore in quanto il ripristino è semplificato. Il processo di modifica comporta un rischio ridotto e l'impatto di una modifica non corretta è ridotto.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Applica modifiche frequenti, minime e reversibili per ridurre la portata e l'impatto di una modifica. In questo modo si semplifica la risoluzione dei problemi, si velocizza la correzione ed è possibile eseguire il rollback di una modifica. Inoltre, aggiunge più rapidamente valore al business.

Risorse

Best practice correlate:

- OPS05-BP03 Utilizzo di sistemi di gestione delle configurazioni
- OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione
- OPS06-BP04 Automazione dei test e del rollback

Documenti correlati:

- Implementazione di microservizi in AWS
- Microservices - Observability

OPS05-BP10 Automazione completa dell'integrazione e della distribuzione

Automatizza la creazione, la distribuzione e il test del carico di lavoro. Questo riduce gli errori causati dai processi manuali e l'impegno necessario per distribuire le modifiche.

Applica i metadati utilizzando i tag delle risorse e AWS Resource Groups seguendo una strategia di applicazione dei tag per agevolare l'identificazione delle risorse. Applica tag alle risorse per organizzare, monitorare i costi e controllare gli accessi e ottimizza l'esecuzione delle attività operative automatizzate.

Risultato desiderato: Chi si occupa di sviluppo utilizza strumenti per distribuire codice ed effettuare la promozione a produzione. Gli sviluppatori non devono effettuare il login alla AWS Management

Console per fornire gli aggiornamenti. Esiste un audit trail completo di modifiche e configurazioni che soddisfa le esigenze di governance e conformità. I processi sono ripetibili e standardizzati tra i team. Gli sviluppatori sono liberi di concentrarsi sullo sviluppo e sui rilasci del codice, aumentando la produttività.

Anti-pattern comuni:

- Venerdì termini la creazione del nuovo codice per il ramo delle funzionalità. Lunedì, dopo aver eseguito gli script di test di qualità del codice e tutti gli script dei test di unità, effettui il check-in del codice per il prossimo rilascio programmato.
- Ti verrà assegnato di codificare una correzione per un problema critico che interessa un numero elevato di clienti nella produzione. Dopo aver testato la correzione, esegui il commit del codice e richiedi via e-mail alla gestione delle modifiche l'approvazione per implementarlo in produzione.
- In qualità di sviluppatore, accedi alla AWS Management Console per creare un nuovo ambiente di sviluppo utilizzando metodi e sistemi non standard.

Vantaggi dell'adozione di questa best practice: Implementando sistemi di gestione automatizzati di compilazione e implementazione, si riduce il numero di errori causati dai processi manuali e lo sforzo di distribuire le modifiche aiutando i membri del team a concentrarsi sull'offerta di valore aggiunto. Maggiore velocità di consegna man mano che procedi verso la promozione a produzione.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Utilizza i sistemi di gestione della compilazione e implementazione per tenere traccia e implementare le modifiche, ridurre gli errori causati dai processi manuali e ridurre il livello di impegno richiesto. Automatizza completamente la pipeline di integrazione e distribuzione dal check-in del codice fino alle fasi di creazione, test, distribuzione e convalida. In questo modo è possibile diminuire il lead time, incoraggiare una maggiore frequenza di modifica, ridurre il livello di impegno e accelerare il time-to-market, il che si traduce in una maggiore produttività e in un aumento della sicurezza del codice man mano che procedi con la promozione verso la produzione.

Risorse

Best practice correlate:

- OPS05-BP03 Utilizzo di sistemi di gestione delle configurazioni

- OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione

Documenti correlati:

- Che cos'è AWS CodeBuild?
- Che cos'è AWS CodeDeploy?

Video correlati:

- AWS re\:Invent 2022 - AWS Well-Architected best practices for DevOps on AWS

OPS 6. In che modo mitighi i rischi della distribuzione?

Adotta prassi per fornire un feedback rapido sulla qualità e che permettano un ripristino veloce dalle modifiche che non hanno i risultati previsti. L'uso di queste prassi consente di mitigare l'impatto dei problemi introdotti attraverso la distribuzione delle modifiche.

Best practice

- OPS06-BP01 Preparazione di un piano in caso di esito negativo delle modifiche
- OPS06-BP02 Implementazioni dei test
- OPS06-BP03 Utilizza strategie di deployment sicure
- OPS06-BP04 Automazione dei test e del rollback

OPS06-BP01 Preparazione di un piano in caso di esito negativo delle modifiche

Pianifica il ripristino di uno stato corretto noto o la correzione nell'ambiente di produzione nel caso in cui l'implementazione generi un risultato indesiderato. Disporre di una politica per stabilire un piano di questo tipo aiuta tutti i team a sviluppare strategie di ripristino dalle modifiche con esito negativo. Alcune strategie di esempio sono le fasi di deployment e rollback, le politiche di modifica, i flag di funzionalità, l'isolamento del traffico e lo spostamento del traffico. Una singola release può includere più modifiche ai componenti correlati. La strategia dovrebbe fornire la capacità di resistere o ripristinare in caso di guasto generato da qualsiasi modifica dei componenti.

Risultato desiderato: hai preparato un piano di ripristino dettagliato per la modifica in caso di fallimento. Inoltre, hai ridotto le dimensioni della release per ridurre al minimo il potenziale impatto su altri componenti del carico di lavoro. Di conseguenza, hai ridotto l'impatto aziendale abbreviando

i potenziali tempi di inattività causati da una modifica non riuscita e aumentando la flessibilità e l'efficienza dei tempi di ripristino.

Anti-pattern comuni:

- Hai eseguito un deployment e l'applicazione è diventata instabile, ma sembra che ci siano utenti attivi sul sistema. Devi decidere se eseguire il rollback della modifica e influire sugli utenti attivi o aspettare di eseguire il rollback della modifica, sapendo che gli utenti potranno essere comunque influenzati.
- Dopo aver apportato una modifica di routine, i nuovi ambienti sono accessibili, ma una delle sottoreti è diventata irraggiungibile. Devi decidere se eseguire il rollback di tutto o provare a correggere il problema della sottorete inaccessibile. Mentre prendi tale decisione, la sottorete rimane irraggiungibile.
- I tuoi sistemi non sono progettati in modo da consentire loro di essere aggiornati con release più piccole. Di conseguenza, è difficile annullare tali modifiche di massa (bulk changes) durante un deployment conclusosi con esito negativo.
- Non utilizzi l'Infrastruttura come codice (IaC) e hai apportato aggiornamenti manuali all'infrastruttura che hanno portato a configurazioni indesiderate. Non è possibile tracciare e ripristinare in modo efficace le modifiche manuali.
- Poiché non hai misurato l'aumento della frequenza dei deployment, il tuo team non è incentivato a ridurre le dimensioni delle modifiche e a migliorare i piani di rollback per ogni modifica, con conseguente aumento dei rischi e dei tassi di fallimento.
- Non misura la durata totale di un'interruzione causata da modifiche con esito negativo. Il tuo team non è in grado di stabilire le priorità e migliorare il processo di deployment e l'efficacia del piano di ripristino.

Vantaggi dell'adozione di questa best practice: Avere un piano per il ripristino dopo le modifiche non riuscite riduce al minimo il tempo medio di ripristino (MTTR) e riduce l'impatto sull'azienda.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Una politica e una pratica coerenti e documentate adottate dai team di rilascio consentono a un'organizzazione di pianificare cosa dovrebbe succedere in caso di modifiche con esito negativo. In circostanze specifiche la politica dovrebbe consentire la possibilità di apportare correzioni per garantire la prosecuzione del processo. In entrambe le situazioni, un piano di correzione (fix forward)

o ripristino (rollback) deve essere ben documentato e testato prima dell'implementazione nei sistemi di produzione live, in modo da ridurre al minimo il tempo necessario per ripristinare una modifica.

Passaggi dell'implementazione

1. Documenta le politiche che richiedono ai team di disporre di piani efficaci per invertire le modifiche entro un periodo di tempo specificato.
   a. Le politiche devono specificare quando è consentita una situazione di applicazione di correzioni per garantire la prosecuzione del processo.
   b. Richiedi un piano di rollback documentato che sia accessibile a tutti i soggetti coinvolti.
   c. Specifica i requisiti per il rollback (ad esempio, quando si rileva che sono state implementate modifiche non autorizzate).
2. Analizza il livello di impatto di tutte le modifiche relative a ciascun componente di un carico di lavoro.
   a. Consenti che le modifiche ripetibili siano standardizzate, basate su modelli e preautorizzate se seguono un flusso di lavoro coerente che applica le politiche di modifica.
   b. Riduci il potenziale impatto di qualsiasi modifica riducendone le dimensioni, in modo che il ripristino richieda meno tempo e abbia un impatto minore sulle attività aziendali.
   c. Assicurati che le procedure di rollback riportino il codice allo stato corretto noto per evitare incidenti, ove possibile.
3. Integra strumenti e flussi di lavoro per applicare le tue politiche in modo programmatico.
4. Rendi visibili i dati sulle modifiche agli altri responsabili di carichi di lavoro per migliorare la velocità di diagnosi di eventuali modifiche con esito negativo che non possono essere ripristinate.
   a. Misura il successo di questa pratica utilizzando dati di modifica visibili e identifica miglioramenti iterativi.
5. Utilizza gli strumenti di monitoraggio per verificare il successo o il fallimento di un deployment per accelerare il processo decisionale sul rollback.
6. Misura la durata dell'interruzione durante una modifica con esito negativo per migliorare continuamente i tuoi piani di ripristino.

Livello di impegno per il piano di implementazione: Medio

Risorse

Best practice correlate:

- OPS06-BP04 Automazione dei test e del rollback

Documenti correlati:

- AWS Builders Library | Ensuring Rollback Safety During Deployments
- AWS Whitepaper | Change Management in the Cloud

Video correlati:

- re:Invent 2019 | Amazon’s approach to high-availability deployment

OPS06-BP02 Implementazioni dei test

Testa le procedure di rilascio in pre-produzione utilizzando la stessa configurazione di deployment, i controlli di sicurezza, i passaggi e le procedure utilizzati nell'ambiente di produzione. Verifica che tutte le fasi implementate siano state completate come previsto, ad esempio l'ispezione di file, configurazioni e servizi. Verifica ulteriormente tutte le modifiche con test funzionali, di integrazione e di carico, oltre ad attivare tutte le attività di monitoraggio come i controlli dell'integrità. Eseguendo questi test, è possibile identificare tempestivamente i problemi di deployment con l'opportunità di pianificarli e mitigarli prima del passaggio nell'ambiente di produzione.

Puoi creare ambienti paralleli temporanei per testare ogni modifica. Automatizza il deployment degli ambienti di test utilizzando l'Infrastruttura come codice (IaC) per ridurre la quantità di lavoro necessaria e garantire stabilità, coerenza e una distribuzione più rapida delle funzionalità.

Risultato desiderato: La tua organizzazione adotta una cultura di sviluppo che include il test delle implementazioni. Ciò garantisce che i team siano concentrati sulla realizzazione di valore aziendale anziché sulla gestione delle release. I team vengono coinvolti fin dall'identificazione dei rischi di deployment per determinare il percorso di mitigazione appropriato.

Anti-pattern comuni:

- Durante le release di produzione, le implementazioni non testate causano problemi frequenti che richiedono una risoluzione mirata e l'escalation.
- La tua release contiene porzioni di Infrastruttura come codice (IaC) che aggiornano le risorse esistenti. Non sei sicuro che l'IaC funzionerà correttamente e non avrà un impatto sulle risorse.
- Viene implementata una nuova funzionalità interessante nella tua applicazione. Non funziona come previsto e non c'è visibilità finché non viene segnalata dagli utenti interessati.

- I certificati vengono aggiornati. Si installano accidentalmente i certificati sui componenti sbagliati, il che non viene rilevato e influisce sui visitatori poiché non è possibile stabilire una connessione sicura al sito web.

Vantaggi dell'adozione di questa best practice: Test approfonditi in fase di pre-produzione delle procedure di implementazione e delle modifiche da queste introdotte riducono al minimo il potenziale impatto sulla produzione causato dalle fasi di implementazione. Ciò aumenta la fiducia durante il rilascio in produzione e riduce al minimo la necessità di supporto operativo senza rallentare la velocità di distribuzione delle modifiche apportate.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Testare il processo di deployment è importante quanto testare le modifiche derivanti dal deployment. Ciò può essere ottenuto testando le fasi di deployment in un ambiente di pre-produzione che rispecchi il più fedelmente possibile quello di produzione. I problemi più comuni, come fasi di deployment incomplete o contenenti errori o configurazioni errate, possono essere individuati di conseguenza prima di passare all'ambiente di produzione. Inoltre, è possibile testare le fasi di ripristino.

Esempio del cliente

Nell'ambito della sua pipeline di integrazione continua e distribuzione continua (CI/CD), AnyCompany Retail esegue i passaggi definiti necessari per rilasciare aggiornamenti dell'infrastruttura e del software per i propri clienti in un ambiente simile a quello di produzione. La pipeline comprende controlli preliminari per rilevare il “drift” (il rilevamento delle modifiche alle risorse eseguite al di fuori dell'IaC) nelle risorse prima del deployment, nonché per convalidare le azioni che l'IaC intraprende al suo avvio. Convalida le fasi di deployment, ad esempio la verifica che determinati file e configurazioni siano presenti e che i servizi siano in esecuzione e rispondano correttamente ai controlli di integrità sull'host locale, prima di effettuare nuovamente la registrazione sul sistema di bilanciamento del carico. Inoltre, tutte le modifiche attivano una serie di test automatici, come test funzionali, di sicurezza, di regressione, di integrazione e di carico.

Passaggi dell'implementazione

1. Esegui controlli di pre-installazione per rispecchiare l'ambiente di pre-produzione in produzione.
   a. utilizza il rilevamento della deviazione per rilevare quando le risorse sono state modificate all'esterno di AWS CloudFormation.

   b. utilizza i set di modifiche per verificare che l'intento dell'aggiornamento dello stack corrisponda alle azioni intraprese da AWS CloudFormation quando viene avviato il set di modifiche.
2. Ciò attiva una fase di approvazione manuale in AWS CodePipeline per autorizzare l'implementazione nell'ambiente di preproduzione.
3. Utilizza configurazioni di implementazione come file AWS CodeDeploy AppSpec per definire le fasi di implementazione e convalida.
4. Ove applicabile, integra AWS CodeDeploy con altri servizi AWS o integra AWS CodeDeploy con prodotti e servizi dei partner.
5. Monitora le implementazioni usando le notifiche di eventi Amazon CloudWatch, AWS CloudTrail e Amazon SNS.
6. Esegui test automatici post-deployment, inclusi test funzionali, di sicurezza, di regressione, di integrazione e di carico.
7. Risoluzione dei problemi di implementazione.
8. La corretta convalida dei passaggi precedenti dovrebbe attivare un flusso di lavoro di approvazione manuale per autorizzare l'implementazione nell'ambiente di produzione.

Livello di impegno per il piano di implementazione: alto

Risorse

Best practice correlate:

- OPS05-BP02 Test e convalida delle modifiche

Documenti correlati:

- AWS Builders' Library | Automating safe, hands-off deployments | Test Deployments
- AWS Whitepaper | Practicing Continuous Integration and Continuous Delivery on AWS
- The Story of Apollo - Amazon's Deployment Engine
- Come eseguire test e debug con AWS CodeDeploy in locale prima di distribuire il codice
- Integrating Network Connectivity Testing with Infrastructure Deployment

Video correlati:

- re:Invent 2020 | Testing software and systems at Amazon

Esempi correlati:

- Tutorial | Deploy and Amazon ECS service with a validation test

OPS06-BP03 Utilizza strategie di deployment sicure

I roll-out sicuri della produzione controllano il flusso di modifiche vantaggiose con l'obiettivo di ridurre al minimo l'impatto percepito di tali modifiche sui clienti. I controlli di sicurezza forniscono meccanismi di ispezione per convalidare i risultati desiderati e limitare l'ambito di impatto derivante da eventuali difetti introdotti dalle modifiche o da errori di deployment. I roll-out sicuri possono includere strategie come feature-flags, one-box, roll-out (release canary), immutabili, suddivisioni del traffico e deployment blu/verdi.

Risultato desiderato: l'organizzazione utilizza un sistema di distribuzione e integrazione continua (CI/CD) che fornisce funzionalità per automatizzare roll-out sicuri. I team sono tenuti a utilizzare strategie di roll-out sicure appropriate.

Anti-pattern comuni:

- Distribuisci una modifica non riuscita a tutta la produzione contemporaneamente. Di conseguenza, tutti i clienti vengono colpiti contemporaneamente.
- Un difetto introdotto in un deployment simultaneo su tutti i sistemi richiede una release di emergenza. La correzione per tutti i clienti richiede diversi giorni.
- La gestione della release di produzione richiede la pianificazione e la partecipazione di diversi team. Ciò limita la tua capacità di aggiornare frequentemente le funzionalità per i tuoi clienti.
- Esegui un deployment variabile modificando i sistemi esistenti. Dopo aver scoperto che la modifica non è andata a buon fine, devi modificare nuovamente i sistemi per ripristinare la versione precedente estendendo il tempo di ripristino.

Vantaggi dell'adozione di questa best practice: Le implementazioni automatizzate bilanciano la velocità dei roll-out con la fornitura costante di modifiche vantaggiose per i clienti. La limitazione dell'impatto previene costosi errori di deployment e massimizza la capacità dei team di rispondere in modo efficiente ai guasti.

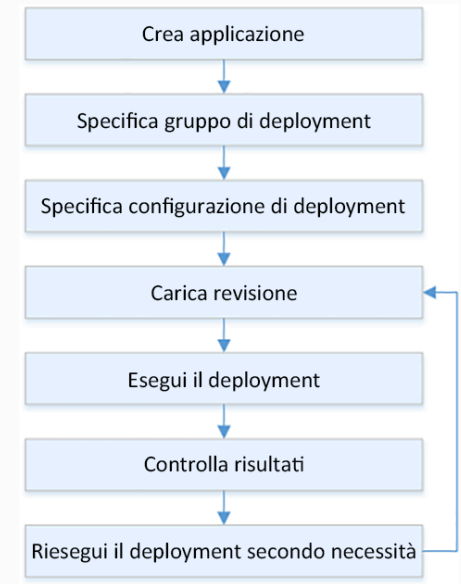
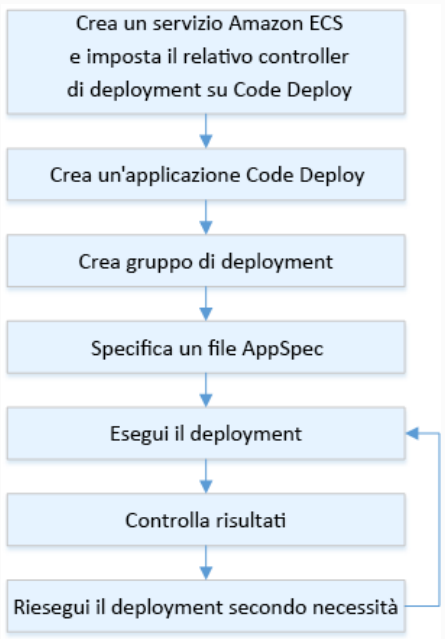
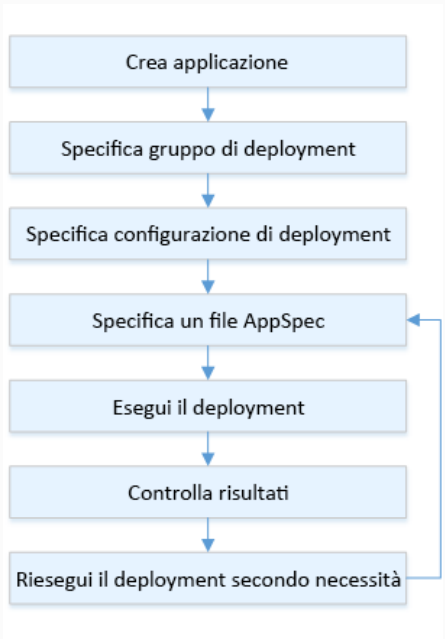
Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Gli errori della distribuzione continua possono portare a una ridotta disponibilità del servizio e a esperienze dei clienti negative. Per massimizzare il tasso di deployment di successo, implementa i controlli di sicurezza nel processo di rilascio end-to-end per ridurre al minimo gli errori di deployment, con l'obiettivo di raggiungere il traguardo di zero errori.

Esempio del cliente

La missione di AnyCompany Retail è raggiungere deployment con tempi di inattività minimi o pari a zero, il che significa che non vi deve essere alcun impatto percepibile dagli utenti durante i deployment. A tal fine, l'azienda ha stabilito modelli di deployment (vedere il seguente diagramma del flusso di lavoro) come roll-out e deployment blu/verdi. Tutti i team adottano uno o più di questi modelli nella loro pipeline CI/CD.

flusso di lavoro CodeDeploy per Amazon EC2	flusso di lavoro CodeDeploy per Amazon ECS	flusso di lavoro CodeDeploy per Lambda
		

Passaggi dell'implementazione

1. Utilizza un flusso di lavoro di approvazione per avviare la sequenza delle fasi di roll-out della produzione al momento della promozione alla produzione.

2. Utilizza un sistema di implementazione automatizzato come AWS CodeDeploy. AWS CodeDeploy opzioni di implementazione include le implementazioni locali (in-place) per EC2/on-premise e le implementazioni blu/verdi per EC2/on-premise AWS Lambda e Amazon ECS (vedi il diagramma del flusso di lavoro precedente).
   a. Ove applicabile, integra AWS CodeDeploy con altri servizi AWS o integra AWS CodeDeploy con prodotti e servizi dei partner.
3. Utilizza implementazioni blu/verdi per database come Amazon Aurora e Amazon RDS.
4. Monitor deployments using Amazon CloudWatch, AWS CloudTrail, and Amazon Simple Notification Service (Amazon SNS) event notifications.
5. Esegui test automatici post-implementazione, inclusi test funzionali, di sicurezza, di regressione, di integrazione e di carico.
6. Risoluzione dei problemi di implementazione.

Livello di impegno per il piano di implementazione: Medio

Risorse

Best practice correlate:

- OPS05-BP02 Test e convalida delle modifiche
- OPS05-BP09 Applicazione di modifiche frequenti, minime e reversibili
- OPS05-BP10 Automazione completa dell'integrazione e della distribuzione

Documenti correlati:

- AWS Builders Library | Automating safe, hands-off deployments | Production deployments
- AWS Builders Library | My CI/CD pipeline is my release captain | Safe, automatic production releases
- AWS Whitepaper | Practicing Continuous Integration and Continuous Delivery on AWS | Deployment methods
- Guida per l'utente di AWS CodeDeploy
- Utilizzo di configurazioni di distribuzione in AWS CodeDeploy
- Configurazione della distribuzione di una release Canary di API Gateway
- Amazon ECS Deployment Types

- Fully Managed Blue/Green Deployments in Amazon Aurora and Amazon RDS
- Distribuzioni blu/verde con AWS Elastic Beanstalk

Video correlati:

- re:Invent 2020 | Hands-off: Automating continuous delivery pipelines at Amazon
- re:Invent 2019 | Amazon's Approach to high-availability deployment

Esempi correlati:

- Prova un'implementazione blu/verde di esempio in AWS CodeDeploy
- Workshop | Buiding CI/CD pipelines for Lambda canary deployments using AWS CDK
- Workshop | Blue/Green and Canary Deployment for EKS and ECS
- Workshop | Building a Cross-account CI/CD Pipeline

OPS06-BP04 Automazione dei test e del rollback

Per aumentare la velocità, l'affidabilità e la sicurezza del processo di deployment, rendi disponibile una strategia per le funzionalità di test e rollback automatizzate negli ambienti di pre-produzione e produzione. Automatizza i test durante il deployment in produzione per simulare le interazioni umane e di sistema che verificano le modifiche implementate. Automatizza il rollback per tornare rapidamente allo stato precedente corretto noto. Il rollback deve essere avviato automaticamente in condizioni predefinite, ad esempio quando il risultato desiderato della modifica non viene raggiunto o quando il test automatico fallisce. L'automazione di queste due attività migliora la percentuale di successo dei deployment, riduce al minimo i tempi di ripristino e riduce il potenziale impatto sulle attività aziendali.

Risultato desiderato: I test automatici e le strategie di rollback sono integrati nella pipeline di integrazione continua e distribuzione continua (CI/CD). Il monitoraggio è in grado di eseguire la convalida in base ai criteri di successo e avviare il rollback automatico in caso di errore. Ciò riduce al minimo l'impatto sugli utenti finali e sui clienti. Ad esempio, quando tutti i risultati dei test sono stati soddisfatti, promuovi il codice nell'ambiente di produzione in cui vengono avviati i test di regressione automatizzati, sfruttando gli stessi casi di test. Se i risultati dei test di regressione non corrispondono alle aspettative, viene avviato il rollback automatico nel flusso di lavoro della pipeline.

Anti-pattern comuni:

- I tuoi sistemi non sono progettati in modo da consentire loro di essere aggiornati con release più piccole. Di conseguenza, è difficile annullare tali modifiche di massa (bulk changes) durante un deployment conclusosi con esito negativo.
- Il processo di deployment consiste in una serie di passaggi manuali. Dopo aver distribuito le modifiche al carico di lavoro, inizi i test post-deployment. Dopo il test, ti rendi conto che il tuo carico di lavoro è inutilizzabile e i clienti sono disconnessi. Inizi quindi a eseguire il rollback alla versione precedente. Tutti questi passaggi manuali ritardano il ripristino complessivo del sistema e provocano un impatto prolungato sui clienti.
- Hai impiegato del tempo a sviluppare casi di test automatizzati per funzionalità che non vengono utilizzate frequentemente nella tua applicazione, riducendo al minimo il ritorno sull'investimento nella tua capacità di eseguire test automatizzati.
- La versione è composta da applicazioni, infrastrutture, patch e aggiornamenti di configurazione indipendenti l'uno dall'altro. Tuttavia, è disponibile un'unica pipeline CI/CD che fornisce tutte le modifiche contemporaneamente. Un guasto in un componente obbliga a ripristinare tutte le modifiche, rendendo il rollback complesso e inefficiente.
- Il tuo team completa il lavoro di codifica nello sprint uno e inizia il lavoro dello sprint due, ma il tuo piano non includeva i test fino allo sprint tre. Come conseguenza, i test automatici hanno rivelato difetti dello sprint uno che dovevano essere risolti prima di poter avviare il test dei deliverable dello sprint due e l'intera release viene ritardata, rendendo inutili i test automatizzati.
- I casi di test di regressione automatizzati per la release di produzione sono completi, ma non stai monitorando lo stato del carico di lavoro. Poiché non è possibile verificare se il servizio è stato riavviato o meno, non sei sicuro se il rollback sia necessario o se sia già avvenuto.

Vantaggi dell'adozione di questa best practice: I test automatizzati aumentano la trasparenza del processo di verifica e la capacità di coprire più funzionalità in un periodo di tempo più breve. Testando e convalidando le modifiche nella produzione, è possibile identificare immediatamente i problemi. Il miglioramento della coerenza con strumenti di test automatizzati consente una migliore rilevazione dei difetti. Effettuando automaticamente il rollback alla versione precedente, l'impatto sui clienti viene ridotto al minimo. In ultima analisi, il rollback automatizzato ispira maggiore fiducia nelle capacità di deployment riducendo l'impatto sulle attività aziendali. Nel complesso, queste funzionalità riducono i tempi di consegna garantendo al contempo la qualità.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Automatizza i test degli ambienti distribuiti per verificare che i risultati siano quelli desiderati. Automatizza il rollback a uno stato corretto noto quando non vengono raggiunti i risultati previsti, per ridurre al minimo il tempo di ripristino e gli errori causati dai processi manuali. Integra gli strumenti di test con il flusso di lavoro della pipeline per testare in modo coerente e ridurre al minimo gli input manuali. Dai priorità all'automazione dei casi di test, come quelli che mitigano i rischi maggiori e devono essere testati frequentemente a ogni modifica. Inoltre, automatizza il rollback in base a condizioni specifiche predefinite nel tuo piano di test.

Passaggi dell'implementazione

1. Stabilisci un ciclo di vita di test per il tuo ciclo di vita di sviluppo che definisca ogni fase del processo di test, dalla pianificazione dei requisiti allo sviluppo dei test case, alla configurazione degli strumenti, ai test automatizzati e alla chiusura dei test case.
   a. Crea un approccio di test specifico per il carico di lavoro partendo dalla tua strategia di test complessiva.
   b. Prendi in considerazione una strategia di test continuo, laddove appropriato, durante tutto il ciclo di vita dello sviluppo.
2. Seleziona strumenti automatizzati per il test e il rollback in base ai requisiti aziendali e agli investimenti nella pipeline.
3. Decidi quali casi di test desideri automatizzare e quali devono essere eseguiti manualmente. Questi possono essere definiti in base alla priorità del valore aziendale della funzionalità testata. Allinea tutti i membri del team su questo piano e verifica la responsabilità per l'esecuzione di test manuali.
   a. Applica le funzionalità di test automatico a casi di test specifici che è opportuno automatizzare, come i casi ripetibili o eseguiti di frequente, quelli che richiedono attività ripetitive o quelli che sono necessari per più configurazioni.
   b. Definisci gli script di automazione dei test e i criteri di successo nello strumento di automazione in modo da poter avviare l'automazione continua del flusso di lavoro quando casi specifici falliscono.
   c. Definisci criteri di errore specifici per il rollback automatico.
4. Dai priorità all'automazione dei test per ottenere risultati coerenti con lo sviluppo accurato e completo di casi di test in cui la complessità e l'interazione umana hanno un rischio maggiore di fallimento.
5. Integra i tuoi strumenti di test e rollback automatizzati nella tua pipeline CI/CD.

   a. Sviluppa criteri di successo chiari per le tue modifiche.
   b. Monitora e osserva per rilevare questi criteri e annullare automaticamente le modifiche quando vengono soddisfatti criteri di rollback specifici.
6. Esegui diversi tipi di test di produzione automatizzati, come:
   a. Test A/B, per mostrare i risultati rispetto alla versione corrente tra due gruppi di utenti di test.
   b. Test Canary, che consente di distribuire la modifica a un sottoinsieme di utenti prima di rilasciarla a tutti.
   c. Test con flag delle funzionalità, che consente di attivare e disattivare una singola funzionalità della nuova versione alla volta dall'esterno dell'applicazione, in modo che ogni nuova funzionalità possa essere convalidata una alla volta.
   d. Test di regressione, per verificare nuove funzionalità con componenti correlati esistenti.
7. Monitora gli aspetti operativi dell'applicazione, delle transazioni e delle interazioni con altre applicazioni e componenti. Sviluppa report per mostrare il successo delle modifiche in base al carico di lavoro in modo da poter identificare quali parti dell'automazione e del flusso di lavoro possono essere ulteriormente ottimizzate.
   a. Sviluppa report sui risultati dei test che ti aiutino a prendere decisioni rapide sull'opportunità o meno di richiamare o meno le procedure di rollback.
   b. Implementa una strategia che consenta il rollback automatico basato su condizioni di errore predefinite derivanti da uno o più metodi di test.
8. Sviluppa i tuoi casi di test automatizzati per consentire la riutilizzabilità in caso di modifiche ripetibili future.

Livello di impegno per il piano di implementazione: Medio

Risorse

Best practice correlate:

- OPS06-BP01 Preparazione di un piano in caso di esito negativo delle modifiche
- OPS06-BP02 Implementazioni dei test

Documenti correlati:

- AWS Builders Library | Ensuring rollback safety during deployments
- Redeploy and rollback a deployment with AWS CodeDeploy

- 8 best practices when automating your deployments with AWS CloudFormation

Esempi correlati:

- Serverless UI testing using Selenium, AWS Lambda, AWS Fargate (Fargate), and AWS Developer Tools

Video correlati:

- re:Invent 2020 | Hands-off: Automating continuous delivery pipelines at Amazon
- re:Invent 2019 | Amazon's Approach to high-availability deployment

OPS 7. Come fai a sapere se hai tutto pronto per supportare un carico di lavoro?

Valuta la disponibilità operativa del carico di lavoro, dei processi e delle procedure, nonché del personale per comprendere i rischi operativi correlati al carico di lavoro.

Best practice

- OPS07-BP01 Verifica della capacità del personale
- OPS07-BP02 Revisione costante della prontezza operativa
- OPS07-BP03 Utilizzo di runbook per eseguire le procedure
- OPS07-BP04 Utilizzo dei playbook per analizzare i problemi
- OPS07-BP05 Adozione di decisioni informate per implementare sistemi e modifiche
- OPS07-BP06 Abilitazione dei piani di supporto per i carichi di lavoro di produzione

OPS07-BP01 Verifica della capacità del personale

Predisponi un meccanismo per stabilire se possiedi il numero appropriato di risorse del personale qualificate per supportare il carico di lavoro. Le risorse devono essere state formate sulla piattaforma e sui servizi che costituiscono il tuo carico di lavoro. Fornisci loro le informazioni necessarie per eseguire il carico di lavoro. Devi avere a disposizione personale qualificato sufficiente per supportare il normale funzionamento del carico di lavoro e gestire gli eventuali incidenti. Predisponi personale sufficiente per la rotazione durante la reperibilità e le ferie per evitare motivi di frustrazione.

Risultato desiderato:

- Presenza di personale qualificato sufficiente per supportare il carico di lavoro nei momenti in cui è disponibile.
- Capacità di fornire al personale formazione sul software e sui servizi che costituiscono il carico di lavoro.

Anti-pattern comuni:

- Implementazione di un carico di lavoro senza membri del team qualificati per l'esecuzione della piattaforma e dei servizi in uso.
- Mancanza di personale sufficiente per supportare la reperibilità a rotazione o le richieste di permesso del personale.

Vantaggi dell'adozione di questa best practice:

- Membri del team qualificati costituiscono un supporto efficace al carico di lavoro.
- Con un numero sufficiente di membri del team, puoi supportare il carico di lavoro e la reperibilità a rotazione, riducendo il rischio di frustrazione.

Livello di rischio associato alla mancata adozione di questa best practice: Elevato

Guida all'implementazione

Verifica che sia disponibile personale qualificato sufficiente per supportare il carico di lavoro. Assicurati che il numero di membri del team di cui disponi sia sufficiente a coprire le normali attività operative, inclusa la reperibilità a rotazione.

Esempio del cliente

AnyCompany Retail si assicura che i team che supportano il carico di lavoro includano personale qualificato sufficiente. L'azienda ha al suo interno un numero sufficiente di tecnici per supportare la reperibilità a rotazione. Il personale riceve formazione sul software e sulla piattaforma su cui è basato il carico di lavoro e viene incoraggiato a conseguire certificazioni. Vi è personale sufficiente per permettere alle persone di richiedere permessi di assenza, continuando a supportare il carico di lavoro durante la reperibilità a rotazione.

Passaggi dell'implementazione

1. Assegna un numero adeguato di risorse del personale per eseguire e supportare il carico di lavoro, tenendo conto della reperibilità.
2. Forma il personale sul software e sulle piattaforme che costituiscono il carico di lavoro.
   a. AWS Training and Certification offre una raccolta di corsi su AWS. Sono disponibili corsi gratuiti e a pagamento, online e di persona.
   b. AWS organizza eventi e webinar in cui puoi apprendere da esperti AWS.
3. Valuta regolarmente le dimensioni e le competenze del team in base al mutare delle condizioni operative e del carico di lavoro. Adegua le dimensioni e le competenze del team ai requisiti operativi.

Livello di impegno per il piano di implementazione: elevato L'assunzione e la formazione di un team per supportare il carico di lavoro possono richiedere un impegno significativo, ma assicurano solidi vantaggi a lungo termine.

Risorse

Best practice correlate:

- OPS11-BP04 Gestione delle informazioni – I membri del team devono disporre delle informazioni necessarie per eseguire e supportare il carico di lavoro. La gestione delle informazioni è il fattore chiave a questo scopo.

Documenti correlati:

- Eventi e webinar AWS
- AWS Training and Certification

OPS07-BP02 Revisione costante della prontezza operativa

Usa le revisioni della prontezza operativa (ORR) per verificare che puoi utilizzare il carico di lavoro. ORR è un meccanismo sviluppato da Amazon per verificare che i team possano utilizzare in sicurezza i propri carichi di lavoro. ORR è un processo di revisione e ispezione che utilizza un elenco di controllo per i requisiti. È un'esperienza self-service che i team utilizzano per certificare i propri carichi di lavoro. Le ORR includono le best practice delle lezioni apprese durante gli anni dedicati alla creazione di software.

Un elenco di controllo ORR è composto da suggerimenti sull'architettura, processo operativo, gestione degli eventi e qualità del rilascio. Il nostro processo di correzione dell'errore (CoE, Correction of Error) è uno dei principali fattori trainanti di questi elementi. L'analisi post-incidente deve guidare l'evoluzione della ORR. Una ORR non riguarda solo l'adozione delle best practice, ma anche la prevenzione del ripetersi di eventi già visti. Infine, in una ORR possono essere inclusi anche i requisiti di sicurezza, governance e conformità.

Esegui le ORR prima che un carico di lavoro venga lanciato nella disponibilità generale e quindi durante tutto il ciclo di vita dello sviluppo software. L'esecuzione della ORR prima del lancio aumenta la tua capacità di utilizzare il carico di lavoro in sicurezza. Riesegui periodicamente la ORR sul carico di lavoro per cogliere eventuali scostamenti dalle best practice. Puoi usare gli elenchi di controllo ORR per il lancio di nuovi servizi e le ORR per le revisioni periodiche. In tal modo puoi tenerti aggiornato sulle nuove best practice che emergono e incorporare le lezioni apprese dall'analisi post-incidente. Man mano che l'utilizzo del cloud cresce, puoi creare i requisiti di ORR nella tua architettura come valori predefiniti.

Risultato desiderato: hai un elenco di controllo ORR con le best practice per la tua organizzazione. Le ORR vengono eseguite prima dell'avvio dei carichi di lavoro. Le ORR vengono eseguite periodicamente nel corso del ciclo di vita del carico di lavoro.

Anti-pattern comuni:

- Avvii un carico di lavoro senza sapere se puoi utilizzarlo.
- I requisiti di governance e sicurezza non sono inclusi nella certificazione di un carico di lavoro per l'avvio.
- I carichi di lavoro non vengono rivalutati periodicamente.
- I carichi di lavoro vengono avviati senza le procedure richieste.
- Si osserva la ripetizione di errori con la stessa causa principale in più carichi di lavoro.

Vantaggi dell'adozione di questa best practice:

- I tuoi carichi di lavoro includono le best practice di architettura, processo e gestione.
- Le lezioni apprese sono incorporate nel processo ORR.
- Le procedure richieste sono in atto all'avvio dei carichi di lavoro.
- Le ORR vengono eseguite durante l'intero ciclo di vita del software dei carichi di lavoro.

Livello di rischio se questa best practice non fosse adottata: alto

Guida all'implementazione

Una ORR è composta da un processo e un elenco di controllo. Il processo ORR deve essere adottato dall'organizzazione e supportato da uno sponsor esecutivo. Come minimo, le ORR devono essere eseguite prima che il carico di lavoro venga lanciato nella disponibilità generale. Esegui la ORR durante tutto il ciclo di vita dello sviluppo software per mantenerlo aggiornato con le best practice o i nuovi requisiti. L'elenco di controllo ORR deve includere elementi di configurazione, requisiti di sicurezza e governance e best practice dell'organizzazione. Nel tempo, puoi utilizzare i servizi, come AWS Config, AWS Security Hube AWS Control Tower Guardrailsper creare le best practice dalla ORR nei guardrail per il rilevamento automatico delle best practice.

Esempio del cliente

Dopo diversi incidenti di produzione, AnyCompany Retail ha deciso di implementare un processo ORR. Ha creato un elenco di controllo composto da best practice, requisiti di governance e conformità e lezioni apprese dalle interruzioni. I nuovi carichi di lavoro conducono le ORR prima dell'avvio. Ogni carico di lavoro esegue una ORR annuale con un sottoinsieme di best practice per incorporare nuove best practice e requisiti che vengono aggiunti all'elenco di controllo ORR. Nel tempo, AnyCompany Retail ha utilizzato AWS Config per individuare le best practices, accelerando il processo ORR.

Passaggi dell'implementazione

Per ulteriori informazioni sulle ORR, consulta il whitepaper Operational Readiness Reviews (ORR) (Revisioni della prontezza operativa (ORR)). Fornisce informazioni dettagliate sulla cronologia del processo ORR, su come creare la procedura ORR e su come sviluppare il proprio elenco di controllo ORR. I passaggi seguenti costituiscono una versione abbreviata di quel documento. Per una comprensione approfondita di cosa sono le ORR e di come crearne una, ti consigliamo di leggere il whitepaper.

1. Riunisci gli stakeholder importanti, inclusi i rappresentanti della sicurezza, delle operazioni e dello sviluppo.
2. Chiedi a ogni stakeholder di indicare almeno un requisito. Per la prima iterazione, prova a limitare il numero di elementi a trenta al massimo.
   - Appendix B: Example ORR questions (Appendice B: Domande ORR di esempio) del whitepaper Operational Readiness Reviews (ORR) (Revisioni della prontezza operativa (ORR)) contiene domande di esempio che puoi utilizzare per iniziare.

3. Raccogli i tuoi requisiti in un foglio di calcolo.
    - Puoi utilizzare gli obiettivi personalizzati nella funzione AWS Well-Architected Tool per sviluppare la ORR e condividerla tra i tuoi account e l'organizzazione AWS.
4. Identifica un carico di lavoro su cui condurre la ORR. L'ideale è un carico di lavoro pre-lancio o un carico di lavoro interno.
5. Scorri l'elenco di controllo ORR e prendi nota di tutti i rilevamenti fatti. I rilevamenti potrebbero non essere validi se è in atto una mitigazione. Aggiungi qualsiasi rilevamento privo di mitigazione al tuo backlog di elementi e implementalo prima del lancio.
6. Continua ad aggiungere le best practice e i requisiti all'elenco di controllo ORR nel corso del tempo.

I clienti di AWS Support con supporto Enterprise possono richiedere il workshop Operational Readiness Review (Revisione sulla prontezza operativa) al proprio Technical Account Manager (TAM). Il workshop è una sessione interattiva di lavoro a ritroso per sviluppare il tuo elenco di controllo ORR.

Livello di impegno per il piano di implementazione: alto. L'adozione di una procedura ORR nella tua organizzazione richiede la sponsorizzazione dell'esecutivo e l'adesione degli stakeholder. Crea e aggiorna l'elenco di controllo con input provenienti da tutta l'organizzazione.

Risorse

Best practice correlate:

- OPS01-BP03 Valutazione dei requisiti di governance - I requisiti di governance sono una scelta naturale per un elenco di controllo ORR.
- OPS01-BP04 Valutazione dei requisiti di conformità - I requisiti di conformità sono talvolta inclusi in un elenco di controllo ORR. Altre volte costituiscono un processo separato.
- OPS03-BP07 Fornitura di risorse appropriate ai team - La capacità del team è un buon requisito ORR.
- OPS06-BP01 Preparazione di un piano in caso di esito negativo delle modifiche - Prima di avviare il carico di lavoro, è necessario stabilire un piano di rollback o rollforward.
- OPS07-BP01 Verifica della capacità del personale - Per supportare un carico di lavoro è necessario disporre del personale necessario.
- SEC01-BP03 Identificazione e convalida degli obiettivi di controllo - Gli obiettivi di controllo della sicurezza costituiscono eccellenti requisiti ORR.

- REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati - I piani di ripristino di emergenza sono un buon requisito ORR.
- COST02-BP01 Sviluppo di politiche basate sui requisiti dell'organizzazione - Le policy di gestione dei costi sono utili da includere nell'elenco di controllo ORR.

Documenti correlati:

- AWS Control Tower - Guardrails in AWS Control Tower (Guardrail in AWS Control Tower)
- AWS Well-Architected Tool - Custom Lenses (Obiettivi personalizzati)
- Operational Readiness Review Template by Adrian Hornsby (Modello di revisione della prontezza operativa di Adrian Hornsby)
- Whitepaper Operational Readiness Reviews (ORR) (Revisioni della prontezza operativa (ORR))

Video correlati:

- AWS Supports You | Building an Effective Operational Readiness Review (ORR) (AWS ti supporta | Creazione di un'efficace revisione della prontezza operativa (ORR))

Esempi correlati:

- Sample Operational Readiness Review (ORR) Lens (Esempio di obiettivi per la revisione della prontezza operativa (ORR))

Servizi correlati:

- AWS Config
- AWS Control Tower
- AWS Security Hub
- AWS Well-Architected Tool

OPS07-BP03 Utilizzo di runbook per eseguire le procedure

Un runbook è un processo documentato finalizzato al raggiungimento di un determinato risultato. I runbook sono composti da una serie di passaggi che è necessario eseguire per conseguire un obiettivo. L'uso dei runbook può essere fatto risalire agli albori dell'aviazione. Nelle operazioni cloud,

è possibile utilizzare i runbook per ridurre i rischi e ottenere i risultati desiderati. In estrema sintesi, un runbook è un elenco di controllo da seguire per completare un'attività.

I runbook costituiscono una parte essenziale del funzionamento dei carichi di lavoro. Dall'inserimento di un nuovo membro in un team all'implementazione di una versione principale, i runbook sono processi codificati che garantiscono risultati coerenti indipendentemente da chi li utilizza. I runbook devono essere pubblicati a livello centralizzato e aggiornati in base all'evoluzione del processo. L'aggiornamento dei runbook rappresenta infatti un elemento chiave dell'intero processo di gestione delle modifiche. Devono inoltre includere le linee guida relative a gestione degli errori, strumenti, autorizzazioni, eccezioni ed escalation in caso di problemi.

A mano a mano che l'organizzazione cresce, è consigliabile automatizzare i runbook. Inizia con runbook concisi e di frequente utilizzo. Utilizza un linguaggio di scripting per automatizzare le procedure o semplificarne l'esecuzione. Dopo aver automatizzato i primi runbook, potrai dedicare altro tempo all'automazione dei runbook più complessi. Gradualmente dovrai automatizzare la maggior parte dei runbook.

Risultato desiderato: il team dispone di una raccolta di linee guida dettagliate per l'esecuzione delle attività relative ai carichi di lavoro. I runbook contengono il risultato desiderato, gli strumenti e le autorizzazioni necessari e le istruzioni per la gestione degli errori. Vengono archiviati in una posizione centralizzata (sistema di controllo delle versioni) e aggiornati di frequente. Ad esempio, i runbook forniscono ai team le funzionalità per monitorare, comunicare e rispondere agli eventi AWS Health degli account critici durante gli allarmi delle applicazioni, i problemi operativi e gli eventi del ciclo di vita pianificati.

Anti-pattern comuni:

- Ricorso alla memoria per completare i singoli passaggi di un processo.
- Implementazione manuale delle modifiche senza utilizzare un elenco di controllo.
- Vari membri dei team eseguono lo stesso processo con procedure o risultati diversi.
- Mancato aggiornamento dei runbook in base alle modifiche o ai processi di automazione del sistema.

Vantaggi dell'adozione di questa best practice:

- Riduzione della percentuale degli errori per le attività manuali.
- Le operazioni vengono eseguite in modo coerente.

- I nuovi membri dei team possono essere operativi da subito.
- I runbook possono essere automatizzati per semplificare le operazioni più impegnative.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I runbook possono avere vari formati, a seconda del livello di "maturità" dell'organizzazione. Nella loro formulazione minima, devono essere un documento di testo in cui sono dettagliate le procedure. Il risultato desiderato deve essere indicato in modo chiaro e preciso. Devono inoltre documentare in modo chiaro le autorizzazioni e gli strumenti speciali necessari. Devono includere linee guida dettagliate relative alle gestione degli errori e ai livelli di escalation nel caso in cui si verifichino problemi o errori. I runbook devono riportare il nome del proprietario ed essere pubblicati in una posizione centralizzata. Dopo averlo compilato, un runbook deve essere convalidato. A tale scopo, devi far eseguire il runbook da un membro diverso del tuo team. A mano a mano che la procedura si evolve, aggiorna i runbook in base al processo di gestione delle modifiche.

I runbook in formato testuale devono essere automatizzati a seconda dell'evoluzione dell'organizzazione. Utilizzando servizi come le automazioni AWS Systems Manager, puoi trasformare un testo non formattato in automazioni che possono essere eseguite nell'ambito del carico di lavoro. Queste automazioni possono essere eseguite in risposta a eventi, per ridurre il carico operativo a salvaguardia del carico di lavoro. L'automazione AWS Systems Manager offre anche un'esperienza di progettazione visiva low-code per creare più facilmente i runbook di automazione.

Esempio del cliente

AnyCompany Retail deve eseguire aggiornamenti dello schema del database durante le implementazioni del software. Il team responsabile delle operazioni cloud ha lavorato assieme al team addetto all'amministrazione del database per redigere un runbook per l'implementazione manuale di queste modifiche. Nel runbook sono incluse le procedure dettagliate sotto forma di elenco di controllo. È presente anche una sezione sulla gestione degli errori in caso di problemi. Il runbook è stato pubblicato assieme ad altri runbook sul wiki interno. Il team responsabile delle operazioni cloud pensa di pianificare l'automazione del runbook in futuro.

Passaggi dell'implementazione

Se non è presente un repository di documenti, è consigliabile creare una libreria di runbook utilizzando un repository per il controllo delle versioni. Puoi creare i runbook utilizzando Markdown. Di

seguito è riportato un modello di runbook di esempio che è possibile utilizzare come riferimento per la creazione dei runbook.

```
# Runbook Title
## Runbook Info
| Runbook ID | Description | Tools Used | Special Permissions | Runbook Author | Last
 Updated | Escalation POC |
|-------|-------|-------|-------|-------|-------|-------|
| RUN001 | What is this runbook for? What is the desired outcome? | Tools | Permissions
 | Your Name | 2022-09-21 | Escalation Name |
## Steps
1. Step one
2. Step two
```

1. Se non disponi di un repository o di un wiki per la documentazione, crea un repository per il controllo delle versioni nel sistema di controllo delle versioni in uso.
2. Individua un processo che non ha un runbook. Un processo ideale è un processo eseguito a cadenza più o meno regolare, con un numero limitato di passaggi e con errori a basso impatto.
3. Nel repository di documenti, crea una nuova bozza di documento Markdown utilizzando il modello. Specifica il titolo del runbook e i campi obbligatori in Informazioni runbook.
4. Partendo dal primo passaggio, compila l'area Passaggi del runbook.
5. Associa il runbook a un membro del team. Chiedi a tale membro di utilizzare il runbook per convalidare i passaggi. In caso di informazioni mancanti o poca chiarezza, aggiorna il runbook.
6. Pubblica il runbook nell'archivio della documentazione interna. Comunica l'avvenuta pubblicazione al team e alle altre parti interessate.
7. In questo modo, nel corso del tempo creerai una libreria di runbook. Man mano che la libreria cresce, comincia a pensare di automatizzare i runbook.

Livello di impegno per il piano di implementazione: basso Lo standard minimo previsto per i runbook è una guida dettagliata in formato testo. L'automazione dei runbook può aumentare l'impegno a livello di implementazione.

Risorse

Best practice correlate:

- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure

- OPS07-BP04 Utilizzo dei playbook per analizzare i problemi
- OPS10-BP01 Utilizzo di un processo per la gestione di eventi, incidenti e problemi
- OPS10-BP02 Definizione di un processo per ogni avviso
- OPS11-BP04 Gestione delle conoscenze

Documenti correlati:

- AWS Well-Architected Framework: Concepts: Runbook development
- Achieving Operational Excellence using automated playbook and runbook
- AWS Systems Manager: Working with runbooks
- Migration playbook for AWS large migrations - Task 4: Improving your migration runbooks
- Use AWS Systems Manager Automation runbooks to resolve operational tasks

Video correlati:

- AWS re:Invent 2019: DIY guide to runbooks, incident reports, and incident response
- How to automate IT Operations on AWS | Amazon Web Services
- Integrate Scripts into AWS Systems Manager

Esempi correlati:

- Well-Architected Labs: Automating operations with Playbooks and Runbooks
- AWS Blog Post: Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services
- AWS Systems Manager: Automation walkthroughs
- AWS Systems Manager: Restore a root volume from the latest snapshot runbook
- Building an AWS incident response runbook using Jupyter notebooks and CloudTrail Lake
- Gitlab - Runbooks
- Rubix - A Python library for building runbooks in Jupyter Notebooks
- Using Document Builder to create a custom runbook

Servizi correlati:

- AWS Systems Manager Automation

OPS07-BP04 Utilizzo dei playbook per analizzare i problemi

I playbook sono guide dettagliate che vengono utilizzate quando si verificano incidenti per analizzare, valutare l'impatto e identificare la causa principale del problema. I playbook sono utili in molti scenari diversi, dalle implementazioni non riuscite agli incidenti di sicurezza. In molti casi, i playbook identificano la causa principale che viene poi mitigata tramite un runbook. I playbook costituiscono un componente essenziale dei piani di risposta agli incidenti di ogni organizzazione.

Un buon playbook include diverse funzionalità chiave che guidano l'utente, passo dopo passo, nel processo di rilevamento. Ma quali passaggi deve eseguire l'utente per diagnosticare un incidente? Illustra chiaramente nel playbook se sono necessari strumenti speciali o autorizzazioni elevate. È essenziale predisporre un piano di comunicazione per aggiornare le parti interessate sullo stato dell'analisi. Nelle situazioni in cui non è possibile identificare la causa principale, il playbook deve prevedere un piano di escalation. Se viene identificata la causa principale, il playbook deve includere il riferimento di un runbook che descrive come risolvere il problema. I playbook devono essere archiviati centralmente e aggiornati regolarmente. Se i playbook vengono utilizzati per avvisi specifici, fornisci al team i riferimenti dei playbook all'interno degli avvisi.

Man mano che l'organizzazione acquisisce maturità, puoi automatizzare i playbook. Inizia con i playbook che trattano incidenti a basso rischio. Utilizza gli script per automatizzare i passaggi di rilevamento. Assicurati di avere i relativi runbook per mitigare le cause principali più comuni.

Risultato desiderato: l'organizzazione dispone dei playbook per gli incidenti comuni. I playbook sono archiviati in una posizione centrale e disponibili per i membri del team. I playbook vengono aggiornati frequentemente. Per qualsiasi causa principale nota, vengono creati i relativi runbook.

Anti-pattern comuni:

- Non esiste un modo standard per analizzare un incidente.
- I membri del team confidano nella "memoria muscolare" o nelle conoscenze istituzionali per risolvere i problemi di un'implementazione non riuscita.
- I nuovi membri del team apprendono come analizzare i problemi attraverso tentativi ed errori.
- Le best practice per l'analisi dei problemi non sono condivise tra i team.

Vantaggi dell'adozione di questa best practice:

- I playbook rendono più efficaci le tue attività per mitigare gli incidenti.
- Uno stesso playbook può essere utilizzato da diversi membri del team in modo da identificare la causa principale in modo coerente.
- Le cause principali note possono già disporre di runbook appositamente sviluppati, accelerando i tempi di ripristino.
- I playbook contribuiscono ad accelerare la collaborazione tra i membri del team.
- I team possono applicare i processi su vasta scala tramite i playbook ripetibili.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Il modo in cui crei e utilizzi i playbook dipende dalla maturità della tua organizzazione. Se non hai familiarità con il cloud, crea i playbook in formato testo in un repository per i documenti centrale. Man mano che l'organizzazione acquisisce maturità, i playbook possono diventare semi automatizzati tramite script scritti in linguaggi come Python. Questi script possono essere eseguiti all'interno di un notebook Jupyter per accelerare il rilevamento. Le organizzazioni avanzate dispongono di playbook completamente automatizzati per i problemi comuni che vengono risolti automaticamente con i runbook.

Inizia a creare i playbook elencando gli incidenti comuni che si verificano nel tuo carico di lavoro. Scegli i playbook per gli incidenti a basso rischio e in cui la causa principale è riconducibile a pochi problemi. Una volta creati i playbook per gli scenari più semplici, passa agli scenari a rischio più elevato o in cui la causa principale non è ancora nota.

I playbook in formato testo vengono automatizzati man mano che l'organizzazione acquisisce maturità. Utilizzando servizi come le automazioni AWS Systems Manager, il testo normale può essere trasformato in automazioni che possono essere eseguite sul carico di lavoro per accelerare le analisi. Queste automazioni possono essere attivate in risposta agli eventi, riducendo il tempo medio per rilevare e risolvere gli incidenti.

I clienti possono utilizzare AWS Systems Manager Incident Manager per rispondere agli incidenti. Questo servizio fornisce un'unica interfaccia per valutare gli incidenti, informare le parti interessate circa il rilevamento e la mitigazione e collaborare per tutta la durata dell'incidente. Utilizza le automazioni AWS Systems Manager per accelerare il rilevamento e il ripristino.

Esempio del cliente

Si è verificato un incidente che ha avuto un impatto sulla produzione della società AnyCompany Retail. L'ingegnere di turno utilizza un playbook per analizzare il problema e man mano che esegue i passaggi, mantiene aggiornati gli stakeholder indicati nel playbook. L'ingegnere identifica la causa principale come una race condition di un servizio di back-end. Utilizzando un runbook, l'ingegnere riavvia il servizio e riporta quindi AnyCompany Retail online.

Passaggi dell'implementazione

Se non è già presente, è consigliabile creare un repository per i documenti con il controllo delle versioni per la libreria di playbook. Puoi creare i tuoi playbook utilizzando Markdown, che è compatibile con la maggior parte dei sistemi di automazione dei playbook. Se parti da zero, utilizza il seguente modello di playbook come esempio.

```
# Playbook Title
## Playbook Info
| Playbook ID | Description | Tools Used | Special Permissions | Playbook Author | Last
 Updated | Escalation POC | Stakeholders | Communication Plan |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| RUN001 | What is this playbook for? What incident is it used for? | Tools |
 Permissions | Your Name | 2022-09-21 | Escalation Name | Stakeholder Name | How will
 updates be communicated during the investigation? |
## Steps
1. Step one
2. Step two
```

1. Se non disponi di un repository o di un wiki per i documenti, crea nel sistema di controllo delle versioni in uso un nuovo repository con il controllo delle versioni per i tuoi playbook.
2. Identifica un problema comune che richieda un'analisi, vale a dire uno scenario in cui la causa principale è riconducibile a pochi problemi e la risoluzione è a basso rischio.
3. Utilizzando il modello Markdown, compila la sezione Titolo del playbook e i campi in Informazioni sul playbook.
4. Includi i passaggi per la risoluzione dei problemi. Illustra nel modo più chiaro possibile le azioni da eseguire o le aree da analizzare.
5. Chiedi a un membro del team di esaminare e convalidare il tuo playbook. Se manca un'informazione o è necessario un chiarimento, aggiorna il playbook.
6. Pubblica il tuo playbook nel repository per i documenti e informa il tuo team e tutte le parti interessate.

7. Questa libreria diventerà sempre più ricca man mano che aggiungi altri playbook. Una volta che sono disponibili diversi playbook, inizia ad automatizzarli con strumenti come le automazioni AWS Systems Manager per mantenere sincronizzati l'automazione e i playbook.

Livello di impegno per il piano di implementazione: basso I playbook sono documenti di testo archiviati in una posizione centrale. Le organizzazioni che hanno acquisito maturità applicano l'automazione dei playbook.

Risorse

Best practice correlate:

- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure
- OPS07-BP03 Utilizzo di runbook per eseguire le procedure
- OPS10-BP01 Utilizzo di un processo per la gestione di eventi, incidenti e problemi
- OPS10-BP02 Definizione di un processo per ogni avviso
- OPS11-BP04 Gestione delle conoscenze

Documenti correlati:

- AWS Well-Architected Framework: Concepts: Playbook development
- Achieving Operational Excellence using automated playbook and runbook
- AWS Systems Manager: Working with runbooks
- Use AWS Systems Manager Automation runbooks to resolve operational tasks

Video correlati:

- AWS re:Invent 2019: DIY guide to runbooks, incident reports, and incident response (SEC318-R1)
- AWS Systems Manager Incident Manager - AWS Virtual Workshops
- Integrate Scripts into AWS Systems Manager

Esempi correlati:

- AWS Customer Playbook Framework
- AWS Systems Manager: Automation walkthroughs

- Building an AWS incident response runbook using Jupyter notebooks and CloudTrail Lake
- Rubix – A Python library for building runbooks in Jupyter Notebooks
- Using Document Builder to create a custom runbook
- Well-Architected Labs: Automating operations with Playbooks and Runbooks
- Well-Architected Labs: Incident response playbook with Jupyter

Servizi correlati:

- AWS Systems Manager Automation
- AWS Systems Manager Incident Manager

OPS07-BP05 Adozione di decisioni informate per implementare sistemi e modifiche

Predisponi processi per la gestione delle modifiche efficaci e infruttuose al carico di lavoro. Si definisce "pre-mortem" un esercizio in cui il team simula un errore per sviluppare strategie di mitigazione. Utilizza questo esercizio per prevedere errori e creare procedure ove opportuno. Valuta i vantaggi e i rischi dell'implementazione di modifiche nel carico di lavoro. Verifica che tutte le modifiche siano conformi ai requisiti di governance.

Risultato desiderato:

- Adozione di decisioni informate durante l'implementazione di modifiche nel carico di lavoro.
- Le modifiche sono conformi ai requisiti di governance.

Anti-pattern comuni:

- Implementazione di una modifica nel carico di lavoro senza un processo per la gestione di un'implementazione errata.
- Applicazione di modifiche all'ambiente di produzione che non sono conformi ai requisiti di governance.
- Implementazione di una nuova versione del carico di lavoro senza stabilire valori di riferimento per l'utilizzo delle risorse.

Vantaggi dell'adozione di questa best practice:

- L'azienda è preparata all'effetto di modifiche infruttuose al carico di lavoro.

- Le modifiche apportate al carico di lavoro sono conformi ai criteri di governance.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

Usa esercizi pre-mortem per sviluppare processi per la gestione di modifiche infruttuose. Documenta i processi di gestione delle modifiche infruttuose. Verifica che tutte le modifiche siano conformi ai requisiti di governance. Valuta i vantaggi e i rischi dell'implementazione di modifiche nel carico di lavoro.

Esempio del cliente

AnyCompany Retail svolge regolarmente esercizi pre-mortem per convalidare i propri processi di gestione delle modifiche infruttuose. L'azienda documenta i propri processi in un Wiki condiviso che aggiorna spesso. Tutte le modifiche sono conformi ai requisiti di governance.

Passaggi dell'implementazione

1. Prendi decisioni informate durante l'implementazione di modifiche nel carico di lavoro. Definisci ed esamina i criteri per un'implementazione corretta. Sviluppa scenari o criteri che attiverebbero il ripristino dello stato precedente a una modifica. Soppesa i vantaggi dell'implementazione di modifiche rispetto ai rischi di una modifica infruttuosa.
2. Verifica che tutte le modifiche siano conformi ai requisiti di governance.
3. Usa esercizi pre-mortem per pianificare la gestione delle modifiche infruttuose e documentare le strategie di mitigazione. Esegui un esercizio di simulazione di un'emergenza per modellare una modifica infruttuosa e convalidare le procedure di ripristino dello stato precedente.

Livello di impegno per il piano di implementazione: moderato. L'implementazione di una procedura di pre-mortem richiede il coordinamento e l'impegno degli stakeholder in tutta l'organizzazione

Risorse

Best practice correlate:

- OPS01-BP03 Valutazione dei requisiti di governance – I requisiti di governance sono un fattore chiave per determinare se implementare una modifica.
- OPS06-BP01 Preparazione di un piano in caso di esito negativo delle modifiche – Predisponi piani per mitigare un'implementazione non riuscita e usa esercizi di pre-mortem per convalidarli.

- OPS06-BP02 Implementazioni dei test – Ogni modifica software deve essere testata nel modo adeguato prima dell'implementazione per ridurre gli errori nell'ambiente di produzione.
- OPS07-BP01 Verifica della capacità del personale – La presenza di personale qualificato sufficiente per supportare il carico di lavoro è essenziale per prendere una decisione informata riguardo all'implementazione di una modifica di sistema.

Documenti correlati:

- Amazon Web Services: rischio e conformità
- Modello di responsabilità condivisa AWS
- Governance nel Cloud AWS: il giusto equilibrio tra agilità e sicurezza

OPS07-BP06 Abilitazione dei piani di supporto per i carichi di lavoro di produzione

Abilita il supporto per qualsiasi software e servizio a cui si affida il tuo carico di lavoro di produzione. Seleziona un livello di supporto adeguato per soddisfare le esigenze di assistenza della produzione. I piani di supporto per queste dipendenze sono necessari nel caso si verifichi un'interruzione del servizio o un problema di software. Documenta i piani di supporto e come chiedere assistenza per tutti i servizi e i fornitori di software. Implementa meccanismi di verifica per controllare che i riferimenti del supporto siano aggiornati.

Risultato desiderato:

- Implementa piani di supporto per software e servizi a cui si affidano i carichi di lavoro di produzione.
- Scegli un piano di supporto adeguato in base alle esigenze di assistenza.
- Documenta i piani e i livelli di supporto e come richiedere assistenza.

Anti-pattern comuni:

- Non hai piani di supporto per un fornitore software strategico. Il tuo carico di lavoro è coinvolto e non puoi fare nulla per accelerare un intervento risolutivo o per ricevere aggiornamenti tempestivi dal fornitore.
- Uno sviluppatore, che era il punto di contatto primario di un fornitore di software, ha lasciato l'azienda. Non puoi contattare direttamente l'assistenza del fornitore. Devi investire il tuo tempo

per cercare le informazioni e orientarti tra sistemi di contatto generici, aumentando così il livello di impegno richiesto per intervenire quando necessario.

- Si verifica un'interruzione della produzione con un fornitore di software. Non esiste una documentazione su come inserire una richiesta di assistenza.

Vantaggi dell'adozione di questa best practice:

- Con il livello di supporto adeguato, puoi ottenere una risposta nei tempi previsti per soddisfare le esigenze in termini di livelli di servizio.
- In caso di problemi in produzione, puoi inoltrare il problema se sei un cliente assistito.
- Fornitori di software e servizi possono essere di aiuto per la risoluzione dei problemi durante un incidente.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Abilita i piani di supporto per qualsiasi fornitore di software e servizi a cui si affida il tuo carico di lavoro di produzione. Configura piani di supporto adeguati per soddisfare le esigenze di assistenza. Per i clienti AWS, questo significa abilitare il supporto Business di AWS o superiore su qualsiasi account su cui hai carichi di lavoro di produzione. Incontra con regolarità i fornitori del servizio di assistenza per ricevere aggiornamenti sulle offerte di supporto, sui processi e sui contatti. Documenta come richiedere assistenza ai fornitori di software e servizi, incluso come inoltrare il problema in caso si verificasse un'interruzione. Implementa meccanismi di aggiornamento dei contatti del supporto.

Esempio del cliente

In AnyCompany Retail, tutte le dipendenze di servizi e software commerciali hanno piani di supporto. Ad esempio, hanno il supporto Enterprise di AWS abilitato su tutti gli account con carichi di lavoro di produzione. In caso di problemi, qualsiasi sviluppatore può inserire una richiesta di assistenza. Esiste una pagina wiki con informazioni su come richiedere assistenza, chi contattare e quali best practice seguire per accelerare il processo di risoluzione.

Passaggi dell'implementazione

1. Lavora con le parti interessate all'interno della tua organizzazione per identificare i fornitori di software e servizi su cui si basa il tuo carico di lavoro. Documenta queste dipendenze.

2. Stabilisci le esigenze in termini di assistenza del tuo carico di lavoro. Seleziona un piano di supporto in linea con tali esigenze.
3. Per software e servizi commerciali definisci un piano di supporto con i fornitori.
    a. Sottoscrivere il supporto Business di AWS o un livello superiore per tutti gli account di produzione garantisce tempi di risposta più rapidi da AWS Support ed è una scelta fortemente consigliata. Se non hai il supporto premium, devi avere un piano di azione per gestire i problemi, che richiede l'aiuto di AWS Support. AWS Support offre un mix di strumenti e tecnologie, persone e programmi progettati per aiutarti in modo proattivo a ottimizzare le performance, ridurre i costi e innovare più rapidamente. Il supporto Business di AWS offre vantaggi aggiuntivi, tra cui l'accesso a AWS Trusted Advisor e ad AWS Personal Health Dashboard, nonché tempi di risposta più rapidi.
4. Documenta il tuo piano di supporto nello strumento di gestione delle conoscenze. Includi come richiedere assistenza, chi avvertire se viene inviata una richiesta di assistenza e come inoltrare il problema durante un incidente. Un wiki è un buon meccanismo che consente a tutti di apportare gli aggiornamenti necessari alla documentazione, nel momento in cui vengono a conoscenza di modifiche a processi o contatti del supporto.

Livello di impegno per il piano di implementazione: Basso. La maggior parte di fornitori di servizi e software offre piani di supporto da attivare. Documentando e condividendo le best practice di supporto sul tuo sistema di gestione delle conoscenze puoi verificare che il tuo team sappia cosa fare quando si verifica un problema in produzione.

Risorse

Best practice correlate:

- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure

Documenti correlati:

- Piani AWS Support

Servizi correlati:

- Supporto del Business AWS
- Supporto Enterprise AWS

Operatività

Domande

- OPS 8. Come utilizzi l'osservabilità del carico di lavoro nella tua organizzazione?
- OPS 9. Come fai a comprendere lo stato delle operazioni?
- OPS 10. In che modo gestisci gli eventi del carico di lavoro e delle operazioni?

OPS 8. Come utilizzi l'osservabilità del carico di lavoro nella tua organizzazione?

Garantisci l'integrità del carico di lavoro sfruttando l'osservabilità. Utilizza metriche, log e tracce pertinenti per ottenere una visione completa delle prestazioni del carico di lavoro e risolvere i problemi in modo efficiente.

Best practice

- OPS08-BP01 Analisi delle metriche del carico di lavoro
- OPS08-BP02 Analizza i log relativi ai carichi di lavoro
- OPS08-BP03 Analisi delle tracce del carico di lavoro
- OPS08-BP04 Creare avvisi fruibili
- OPS08-BP05 Creare dashboard

OPS08-BP01 Analisi delle metriche del carico di lavoro

Dopo aver implementato la telemetria dell'applicazione, analizza regolarmente le metriche raccolte. Sebbene latenza, richieste, errori e capacità (o quote) forniscano informazioni dettagliate sulle prestazioni del sistema, è fondamentale dare priorità alla revisione delle metriche relative ai risultati aziendali. Ciò ti assicura di prendere decisioni basate sui dati in linea con i tuoi obiettivi aziendali.

Risultato desiderato: Informazioni dettagliate sulle prestazioni del carico di lavoro che guidano decisioni basate sui dati, garantendo l'allineamento con gli obiettivi aziendali.

Anti-pattern comuni:

- Analisi isolata delle metriche senza considerare il loro impatto sui risultati aziendali.
- Eccessiva dipendenza dalle metriche tecniche trascurando quelle aziendali.
- Revisione poco frequente delle metriche, perdita di opportunità di prendere decisioni in tempo reale.

Vantaggi dell'adozione di questa best practice:

- Comprensione migliorata della correlazione tra prestazioni tecniche e risultati aziendali.
- Processo decisionale migliorato basato su dati in tempo reale.
- Identificazione e mitigazione proattive dei problemi prima che influiscano sui risultati aziendali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Sfrutta strumenti come Amazon CloudWatch per l'esecuzione di analisi delle metriche. Utilizza servizi AWS come AWS Cost Anomaly Detection e Amazon DevOps Guru per rilevare anomalie, soprattutto quando le soglie statiche non sono conosciute o quando i modelli di comportamento evidenziano possibili anomalie.

Passaggi dell'implementazione

1. Analizza e revisiona: revisiona e interpreta regolarmente le metriche relative al carico di lavoro.
   a. Dai priorità alle metriche relative ai risultati aziendali rispetto a quelle puramente tecniche.
   b. Comprendi l'importanza di picchi, cali o schemi nei dati.
2. Utilizza Amazon CloudWatch: utilizza Amazon CloudWatch per una visualizzazione centralizzata e un'analisi approfondita.
   a. Configura dashboard CloudWatch per visualizzare le tue metriche e confrontarle nel tempo.
   b. Utilizza percentili in CloudWatch per avere una visione chiara della distribuzione delle metriche, il che può aiutarti a definire gli SLA e a identificare valori anomali.
   c. Configura AWS Cost Anomaly Detection per identificare modelli insoliti senza fare affidamento su soglie statiche.
   d. Implementa l'osservabilità CloudWatch tra account per monitorare e risolvere i problemi delle applicazioni che si estendono su più account all'interno di una regione.
   e. Utilizza gli approfondimenti sulle metriche CloudWatch per interrogare e analizzare i dati delle metriche tra account e regioni, identificando tendenze e anomalie.
   f. Applica Metrica matematica CloudWatch per trasformare, aggregare o eseguire calcoli sulle metriche per ottenere informazioni più approfondite.
3. Impiega Amazon DevOps Guru: incorpora Amazon DevOps Guru per il rilevamento delle anomalie basato sul machine learning, che consente di identificare i primi segnali di problemi operativi che riguardano le applicazioni serverless e di correggerli prima che abbiano un impatto sui clienti.

4. Ottimizza in base agli approfondimenti: prendi decisioni informate sulla base dell'analisi delle metriche per adeguare e migliorare i carichi di lavoro.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS04-BP02 Implementazione della telemetria dell'applicazione

Documenti correlati:

- The Wheel Blog - Emphasizing the importance of continually reviewing metrics
- Percentile are important
- Utilizzo di AWS Cost Anomaly Detection
- CloudWatch cross-account observability
- Query your metrics with CloudWatch Metrics Insights

Video correlati:

- Enable Cross-Account Observability in Amazon CloudWatch
- Introduction to Amazon DevOps Guru
- Continuously Analyze Metrics using AWS Cost Anomaly Detection

Esempi correlati:

- One Observability Workshop
- Gaining operation insights with AIOps using Amazon DevOps Guru

OPS08-BP02 Analizza i log relativi ai carichi di lavoro

L'analisi regolare dei log dei carichi di lavoro è essenziale per acquisire una comprensione più approfondita degli aspetti operativi dell'applicazione. Attraverso l'analisi, la consultazione e

l'interpretazione efficiente dei dati di log, è possibile ottimizzare continuamente le prestazioni e la sicurezza delle applicazioni.

Risultato desiderato: approfondimenti sul comportamento dell'applicazione e sulle operazioni derivanti da un'analisi completa dei log, che garantisce il rilevamento e la mitigazione proattiva dei problemi.

Anti-pattern comuni:

- Si trascura l'analisi dei log fino a quando non si verifica un problema critico.
- Il mancato utilizzo della suite completa degli strumenti disponibili per l'analisi dei log comporta la perdita di approfondimenti importanti.
- Si fa affidamento esclusivamente sulla revisione manuale dei log senza sfruttare le funzionalità di automazione e query.

Vantaggi dell'adozione di questa best practice:

- Identificazione proattiva dei colli di bottiglia operativi, delle minacce alla sicurezza e di altri problemi potenziali.
- Utilizzo efficiente dei dati di log per l'ottimizzazione continua dell'applicazione.
- Comprensione migliorata del comportamento dell'applicazione, facilitando il debug e la risoluzione dei problemi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Amazon CloudWatch Logs è un potente strumento per l'analisi dei log. Le funzionalità integrate come CloudWatch Logs Insights e Contributor Insights rendono intuitivo ed efficiente il processo di acquisizione di approfondimenti significativi dai log.

Passaggi dell'implementazione

1. Configura CloudWatch Logs: configura applicazioni e servizi per inviare log a CloudWatch Logs.
2. Usa il rilevamento delle anomalie nei log: utilizza il rilevamento delle anomalie Amazon CloudWatch Logs per identificare e avvisare automaticamente se si verificano modelli di log insoliti. Questo strumento consente di gestire in modo proattivo le anomalie nei log e di rilevare tempestivamente i potenziali problemi.

3. Configura CloudWatch Logs Insights: usa CloudWatch Logs Insights per cercare e analizzare in modo interattivo i dati di log.
   a. Crea query per estrarre modelli, visualizzare i dati di log e ricavare approfondimenti utili.
   b. Usa l'analisi dei modelli CloudWatch Logs Insights per analizzare e visualizzare i modelli di log frequenti. Questa funzionalità consente di comprendere le tendenze operative più comuni e i potenziali valori anomali nei dati di log.
   c. Usa il confronto (diff) di CloudWatch Logs per eseguire analisi differenziali tra diversi periodi di tempo o diversi gruppi di log. Questa funzionalità ti consente di individuare le modifiche e valutarne l'impatto sulle prestazioni o sul comportamento del sistema.
4. Monitora i log in tempo reale con Live Tail: usa Amazon CloudWatch Logs Live Tail per visualizzare i dati di log in tempo reale. Puoi monitorare attivamente le attività operative dell'applicazione man mano che si verificano, ottenendo una visibilità immediata sulle prestazioni del sistema e sui potenziali problemi.
5. Utilizza Contributor Insights: utilizza CloudWatch Contributor Insights per identificare i top talker in dimensioni ad alta cardinalità come gli indirizzi IP o gli utenti-agenti.
6. Implementa filtri di metriche CloudWatch Logs: configura i filtri di metriche CloudWatch Logs per convertire i dati di log in metriche fruibili. Puoi così impostare allarmi o analizzare ulteriormente i modelli.
7. Implementa l'osservabilità tra account CloudWatch: monitora e risolvi i problemi delle applicazioni che si estendono su più account all'interno di una regione.
8. Rivedi regolarmente e perfeziona: rivedi periodicamente le tue strategie di analisi dei log per acquisire tutte le informazioni pertinenti e ottimizzare continuamente le prestazioni delle applicazioni.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS04-BP02 Implementazione della telemetria dell'applicazione
- OPS08-BP01 Analisi delle metriche del carico di lavoro

Documenti correlati:

- Analyzing Log Data with CloudWatch Logs Insights
- Using CloudWatch Contributor Insights
- Creating and Managing CloudWatch Log Metric Filters

Video correlati:

- Analyze Log Data with CloudWatch Logs Insights
- Use CloudWatch Contributor Insights to Analyze High-Cardinality Data

Esempi correlati:

- CloudWatch Logs Sample Queries
- One Observability Workshop

OPS08-BP03 Analisi delle tracce del carico di lavoro

L'analisi dei dati di tracciamento è fondamentale per ottenere una visione completa del percorso operativo di un'applicazione. Visualizzando e comprendendo le interazioni tra i vari componenti, è possibile ottimizzare le prestazioni, identificare i colli di bottiglia e migliorare l'esperienza utente.

Risultato desiderato: ottieni una chiara visibilità sulle operazioni distribuite della tua applicazione, che si traduce in una risoluzione più rapida dei problemi e in un'esperienza utente migliorata.

Anti-pattern comuni:

- I dati di tracciamento vengono trascurati e ci si affida esclusivamente a log e metriche.
- I dati di tracciamento non sono correlati ai log associati.
- Vengono ignorate le metriche derivate dalle tracce, come la latenza e i tassi di errore.

Vantaggi dell'adozione di questa best practice:

- Miglioramento della risoluzione dei problemi e riduzione del tempo medio di risoluzione (MTTR).
- Informazioni dettagliate sulle dipendenze e sul loro impatto.
- Identificazione e correzione rapide dei problemi di prestazione.
- Vengono sfruttate le metriche derivate dalle tracce per un processo decisionale informato.
- Esperienze utente migliorate attraverso interazioni con i componenti ottimizzate.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

AWS X-Ray offre una suite completa per l'analisi dei dati di tracciamento, fornendo una visione olistica delle interazioni con i servizi, monitorando le attività degli utenti e rilevando i problemi di prestazioni. Funzionalità come ServiceLens, X-Ray Insights, X-Ray Analytics e Amazon DevOps Guru permettono di ottenere informazioni fruibili più approfondite derivate dai dati di tracciamento.

Passaggi dell'implementazione

I seguenti passaggi offrono un approccio strutturato per implementare efficacemente l'analisi dei dati di tracciamento utilizzando i servizi AWS:

1. Integra AWS X-Ray: assicurati che X-Ray sia integrato con le tue applicazioni per acquisire dati di tracciamento.
2. Analizza le metriche X-Ray: analizza le metriche derivate dalle tracce di X-Ray, come latenza, tassi di richiesta, percentuale di errore e distribuzione dei tempi di risposta utilizzando la mappa dei servizi per monitorare lo stato delle applicazioni.
3. Usa ServiceLens: utilizza la mappa di ServiceLens per ottenere una maggiore osservabilità dei servizi e delle applicazioni. Fornisce la visualizzazione integrata di tracce, metriche, log, allarmi e altre informazioni correlate all'integrità.
4. Abilita X-Ray Insights:
   a. attiva X-Ray Insights per il rilevamento automatico delle anomalie nelle tracce.
   b. Esamina gli approfondimenti per individuare i modelli e determinare le cause ultime, come l'aumento dei tassi di errore o delle latenze.
   c. Consulta la cronologia degli approfondimenti per un'analisi cronologica dei problemi rilevati.
5. Usa X-Ray Analytics: X-Ray Analytics consente di esplorare a fondo i dati di tracciamento, individuare modelli ed estrarre approfondimenti.
6. Usa i gruppi in X-Ray: crea i gruppi in X-Ray per filtrare le tracce in base a criteri come l'elevata latenza, per eseguire un'analisi più mirata.
7. Incorpora Amazon DevOps Guru: integra Amazon DevOps Guru per trarre vantaggio dai modelli di machine learning che individuano le anomalie operative nelle tracce.
8. Usa CloudWatch Synthetics: usa CloudWatch Synthetics per creare canary per il monitoraggio continuo di endpoint e flussi di lavoro. Questi canary possono integrarsi con X-Ray per fornire dati di tracciamento per un'analisi approfondita delle applicazioni testate.

9. Usa Real User Monitoring (RUM): con AWS X-Ray e CloudWatch RUM puoi analizzare ed eseguire il debug del percorso della richiesta a partire dagli utenti finali dell'applicazione tramite servizi AWS gestiti a valle. Ciò ti aiuta a identificare le tendenze e gli errori di latenza che hanno un impatto sugli utenti finali.
10. Esegui correlazioni con i log: esegui correlazioni tra i dati di tracciamento e i relativi log all'interno della vista di tracce di X-Ray per una prospettiva granulare del comportamento delle applicazioni. Ciò consente di visualizzare gli eventi di log associati direttamente alle transazioni tracciate.
11. Implementa l'osservabilità tra account CloudWatch: monitora e risolvi i problemi delle applicazioni che si estendono su più account all'interno di una regione.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS08-BP01 Analisi delle metriche del carico di lavoro
- OPS08-BP02 Analizza i log relativi ai carichi di lavoro

Documenti correlati:

- Using ServiceLens to Monitor Application Health
- Exploring Trace Data with X-Ray Analytics
- Detecting Anomalies in Traces with X-Ray Insights
- Continuous Monitoring with CloudWatch Synthetics

Video correlati:

- Analyze and Debug Applications Using Amazon CloudWatch Synthetics & AWS X-Ray
- Use AWS X-Ray Insights

Esempi correlati:

- One Observability Workshop
- Implementing X-Ray with AWS Lambda
- CloudWatch Synthetics Canary Templates

OPS08-BP04 Creare avvisi fruibili

Rilevare e rispondere tempestivamente alle deviazioni di comportamento dell'applicazione è fondamentale. È importante riconoscere quando i risultati basati sugli indicatori chiave di prestazione (KPI) sono a rischio o quando si verificano anomalie impreviste. Basare gli avvisi sui KPI garantisce che i segnali ricevuti siano direttamente correlati all'impatto aziendale od operativo. Questo approccio verso avvisi fruibili promuove risposte proattive e aiuta a mantenere le prestazioni e l'affidabilità del sistema.

Risultato desiderato: si ricevono avvisi tempestivi, pertinenti e fruibili per l'identificazione e la mitigazione rapida di potenziali problemi, soprattutto quando i risultati dei KPI sono a rischio.

Anti-pattern comuni:

- Si impostano troppi avvisi non critici, con conseguente affaticamento da avvisi ("alert fatigue").
- Non viene data priorità agli avvisi in base ai KPI, il che rende difficile comprendere l'impatto dei problemi sull'azienda.
- Non affrontare le cause principali porta a ricevere avvisi ripetuti per lo stesso problema.

Vantaggi dell'adozione di questa best practice:

- Riduzione dell'affaticamento da avvisi ("alert fatigue") concentrandosi su avvisi pertinenti e fruibili.
- Maggiore operatività e affidabilità del sistema grazie al rilevamento e alla mitigazione proattiva dei problemi.
- Migliore collaborazione tra team e risoluzione più rapida dei problemi grazie all'integrazione con i più diffusi strumenti di avviso e comunicazione.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per creare un meccanismo di avviso efficace, è fondamentale utilizzare metriche, log e dati di tracciamento che segnalino quando i risultati basati sui KPI sono a rischio o vengono rilevate anomalie.

Passaggi dell'implementazione

1. Determina gli indicatori chiave di prestazione (KPI): identifica i KPI dell'applicazione. Gli avvisi devono essere correlati a questi KPI per riflettere accuratamente l'impatto aziendale.

2. Implementa il rilevamento delle anomalie:
    - Usa il rilevamento delle anomalie Amazon CloudWatch: configura il rilevamento delle anomalie Amazon CloudWatch per rilevare automaticamente modelli insoliti e generare avvisi solo per anomalie reali.
    - Utilizza AWS X-Ray Insights:
        a. Configura X-Ray Insights per rilevare anomalie nei dati di tracciamento.
        b. Configura le notifiche per X-Ray Insights per ricevere avvisi quando si rilevano problemi.
    - Esegui l'integrazione con Amazon DevOps Guru:
        a. Utilizza Amazon DevOps Guru e le sue capacità di machine learning per rilevare anomalie operative nei dati esistenti.
        b. Accedi alle impostazioni di notifica in DevOps Guru per configurare gli avvisi per le anomalie.
3. Implementa avvisi fruibili: progetta avvisi che forniscano informazioni adeguate per intraprendere un'azione immediata.
    1. Monitora gli eventi AWS Health con le regole Amazon EventBridge o integra a livello di programmazione l'API AWS Health per automatizzare le azioni quando ricevi eventi AWS Health. Possono essere azioni generali, come l'invio di tutti i messaggi pianificati sugli eventi del ciclo di vita a un'interfaccia di chat, oppure azioni specifiche, come l'avvio di un flusso di lavoro in uno strumento di gestione dei servizi IT.
4. Riduci l'affaticamento da avvisi: riduci al minimo gli avvisi non critici. Quando i team sono sovraccaricati da numerosi avvisi insignificanti, possono trascurare i problemi critici, riducendo l'efficacia complessiva del meccanismo di avviso.
5. Configura allarmi compositi: utilizza gli allarmi compositi Amazon CloudWatch per raggruppare più allarmi.
6. Integra strumenti di avviso: incorpora strumenti come Ops Genie e PagerDuty.
7. Integra AWS Chatbot: integra AWS Chatbot per inoltrare avvisi a Amazon Chime, Microsoft Teams e Slack.
8. Usa l'avviso basato sui log: utilizza i filtri delle metriche dei log in CloudWatch per creare allarmi basati su eventi di log specifici.
9. Rivedi e itera: riesamina e perfeziona regolarmente le configurazioni degli avvisi.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS04-BP02 Implementazione della telemetria dell'applicazione
- OPS04-BP03 Implementazione della telemetria dell'esperienza utente
- OPS04-BP04 Implementazione della telemetria delle dipendenze
- OPS04-BP05 Implementazione del tracciamento distribuito
- OPS08-BP01 Analisi delle metriche del carico di lavoro
- OPS08-BP02 Analizza i log relativi ai carichi di lavoro
- OPS08-BP03 Analisi delle tracce del carico di lavoro

Documenti correlati:

- Using Amazon CloudWatch alarms
- Create a composite alarm
- Create a CloudWatch alarm based on anomaly detection
- DevOps Guru Notifications
- X-ray insights notifications
- Monitor, operate, and troubleshoot your AWS resources with interactive ChatOps
- Amazon CloudWatch Integration Guide | PagerDuty
- Integrate Opsgenie with Amazon CloudWatch

Video correlati:

- Create Composite Alarms in Amazon CloudWatch
- AWS Chatbot Overview
- AWS On Air ft. Mutative Commands in AWS Chatbot

Esempi correlati:

- Alarms, incident management, and remediation in the cloud with Amazon CloudWatch
- Tutorial: Creating an Amazon EventBridge rule that sends notifications to AWS Chatbot

- One Observability Workshop

OPS08-BP05 Creare dashboard

Le dashboard rappresentano la visualizzazione incentrata sull'utente dei dati di telemetria dei carichi di lavoro. Sebbene forniscano un'interfaccia visiva fondamentale, non dovrebbero sostituire i meccanismi di allarme, ma integrarli. Se realizzate con cura, sono in grado di fornire approfondimenti rapidi sullo stato e sulle prestazioni del sistema e possono informare le parti interessate in tempo reale riguardo ai risultati aziendali e all'impatto dei problemi.

Risultato desiderato:

Approfondimenti chiari e fruibili sullo stato del sistema e dell'azienda attraverso rappresentazioni visive.

Anti-pattern comuni:

- Dashboard eccessivamente complicate con troppe metriche.
- Affidarsi a dashboard senza avvisi per il rilevamento delle anomalie.
- Non aggiornare le dashboard man mano che i carichi di lavoro si evolvono.

Vantaggi dell'adozione di questa best practice:

- Visibilità immediata delle metriche e dei KPI critici di sistema.
- Miglioramento della comunicazione e della comprensione con gli stakeholder.
- Approfondimenti rapidi sull'impatto dei problemi operativi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Dashboard incentrate sul business

Le dashboard personalizzate in base ai KPI aziendali coinvolgono una gamma più ampia di parti interessate. Anche se queste persone potrebbero non essere interessate alle metriche di sistema, desiderano comprendere le implicazioni aziendali di questi numeri. Una dashboard incentrata sul business garantisce che tutte le metriche tecniche e operative monitorate e analizzate siano allineate con gli obiettivi aziendali generali. Questo allineamento fornisce chiarezza, garantendo che tutti siano

sulla stessa lunghezza d'onda per quanto riguarda ciò che è essenziale e ciò che non lo è. Inoltre, le dashboard che mettono in evidenza i KPI aziendali tendono ad essere più fruibili. Gli stakeholder possono comprendere rapidamente lo stato delle operazioni, le aree che richiedono attenzione e il potenziale impatto sui risultati aziendali.

Con questo in mente, al momento di creare una dashboard, assicurati che ci sia un equilibrio tra metriche tecniche e KPI aziendali. Entrambi sono fondamentali, ma si rivolgono a un pubblico diverso. Idealmente, dovresti disporre di dashboard che forniscano una visione olistica dello stato e delle prestazioni del sistema, mettendo in evidenza al contempo i principali risultati aziendali e le loro implicazioni.

Le dashboard di Amazon CloudWatch sono home page personalizzabili nella console CloudWatch che puoi utilizzare per monitorare le tue risorse in un'unica visualizzazione, anche quelle distribuite tra Regioni AWS e account diversi.

Passaggi dell'implementazione

1. Crea una dashboard di base: crea una nuova dashboard in CloudWatch, assegnandole un nome descrittivo.
2. Usa i widget Markdown: prima di utilizzare le metriche, usa i widget Markdown per aggiungere un contesto testuale nella parte superiore della dashboard. Questo contesto specifica cosa include la dashboard, qual è l'importanza delle metriche rappresentate e può contenere anche link ad altri dashboard e strumenti di risoluzione dei problemi.
3. Crea le variabili della dashboard: incorpora le variabili della dashboard laddove appropriato per consentire una visualizzazione dinamica e flessibile della dashboard.
4. Crea i widget per le metriche: aggiungi i widget per visualizzare le varie metriche generate dall'applicazione e personalizza questi widget in modo che rappresentino efficacemente lo stato del sistema e i risultati aziendali.
5. Esegui query con Log Insights: utilizza CloudWatch Log Insights per ottenere metriche fruibili dai log e visualizzare questi approfondimenti sulla dashboard.
6. Configura gli allarmi: integra gli allarmi CloudWatch nella tua dashboard per una rapida visualizzazione di tutte le metriche che superano le soglie prestabilite.
7. Usa Contributor Insights: incorpora CloudWatch Contributor Insights per analizzare i campi ad alta cardinalità e comprendere meglio i principali fattori di contribuzione della risorsa.
8. Progetta widget personalizzati: per esigenze specifiche non soddisfatte dai widget standard, valuta la possibilità di creare widget personalizzati. Questi possono attingere da varie origini dati o rappresentare i dati in modi unici.

9. Usa AWS Health Dashboard: utilizza AWS Health Dashboard per ottenere approfondimenti sullo stato del tuo account, sugli eventi e sulle prossime modifiche che potrebbero influire sui servizi e sulle risorse. Puoi anche ottenere una visualizzazione centralizzata degli eventi di integrità in AWS Organizations o creare dashboard personalizzate (per maggiori dettagli, consulta Esempi correlati).
10. Itera e perfeziona: man mano che la tua applicazione si evolve, riesamina regolarmente la dashboard per assicurarne la pertinenza.

Risorse

Best practice correlate:

- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS08-BP01 Analisi delle metriche del carico di lavoro
- OPS08-BP02 Analizza i log relativi ai carichi di lavoro
- OPS08-BP03 Analisi delle tracce del carico di lavoro
- OPS08-BP04 Creare avvisi fruibili

Documenti correlati:

- Building Dashboards for Operational Visibility
- Using Amazon CloudWatch Dashboards

Video correlati:

- Create Cross Account & Cross Region CloudWatch Dashboards
- AWS re:Invent 2021 - Gain enterprise visibility with Cloud AWS operation dashboards)

Esempi correlati:

- One Observability Workshop
- Application Monitoring with Amazon CloudWatch
- AWS Health Events Intelligence Dashboards and Insights
- Visualize AWS Health events using Amazon Managed Grafana

OPS 9. Come fai a comprendere lo stato delle operazioni?

Definisci, acquisisci e analizza i parametri delle operazioni per ottenere visibilità sugli eventi delle operazioni, in modo da intraprendere le azioni appropriate.

Best practice

- OPS09-BP01 Misura gli obiettivi operativi e i KPI con le metriche
- OPS09-BP02 Comunicare lo stato e le tendenze per garantire la visibilità delle operazioni
- OPS09-BP03 Revisione delle metriche operative e assegnazione delle priorità per favorire il miglioramento

OPS09-BP01 Misura gli obiettivi operativi e i KPI con le metriche

Ottieni obiettivi e KPI dalla tua organizzazione che definiscano il successo delle operazioni e stabilisci metriche che li riflettano. Definisci previsioni da utilizzare come riferimento e rivalutale regolarmente. Sviluppa meccanismi per raccogliere queste metriche dai team per la valutazione.

Risultato desiderato:

- Gli obiettivi e i KPI per i team operativi dell'organizzazione sono stati pubblicati e condivisi.
- Vengono stabilite metriche che riflettono questi KPI. Gli esempi possono includere:
  - Lunghezza della coda dei ticket o età media del ticket
  - Numero di ticket raggruppati per tipo di problema
  - Tempo impiegato per lavorare ai problemi con o senza una procedura operativa standardizzata (SOP)
  - Tempo impiegato per il ripristino dopo un push di codice non riuscito
  - Volume delle chiamate

Anti-pattern comuni:

- Le scadenze di implementazione non vengono rispettate perché gli sviluppatori sono costretti a dedicarsi alle attività di risoluzione dei problemi. I team di sviluppo chiedono più personale, ma non possono quantificarne il numero perché il tempo impiegato non può essere misurato.
- È stato installato un desk di livello 1 per gestire le chiamate degli utenti. Nel corso del tempo, sono aumentati i carichi di lavoro ma non il personale assegnato al desk di livello 1. La soddisfazione

dei clienti ne risente a causa dell'aumento dei tempi di chiamata e di quelli per arrivare a una soluzione, ma il team manageriale non vede indicatori di questo problema e non intraprende azioni.

- Un carico di lavoro problematico è stato affidato a un team operativo separato per la gestione. A differenza di altri carichi di lavoro, questo non è accompagnato dalla documentazione e dai runbook adeguati. Pertanto, i team dedicano più tempo alla risoluzione dei problemi e alla gestione degli errori. Tuttavia, non esistono metriche che lo documentino, il che rende difficile comprendere le responsabilità.

Vantaggi dell'adozione di questa best practice: Quando il monitoraggio del carico di lavoro mostra lo stato delle nostre applicazioni e servizi, i team operativi dedicati al monitoraggio forniscono ai proprietari informazioni dettagliate sui cambiamenti avvenuti tra i consumatori di tali carichi di lavoro, come le mutate esigenze aziendali. Misura l'efficacia di questi team e valutali rispetto agli obiettivi aziendali creando metriche in grado di riflettere lo stato delle operazioni. Le metriche possono evidenziare problemi relativi al supporto o identificare quando si verificano deviazioni rispetto a un obiettivo di livello di servizio.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Fissa un appuntamento per parlare con i leader aziendali e gli stakeholder per stabilire gli obiettivi generali del servizio. Stabilisci quali devono essere i compiti dei vari team operativi e quali sfide potrebbero affrontare. Utilizza queste informazioni per un'attività di brainstorming sugli indicatori chiave di prestazione (KPI) che potrebbero riflettere questi obiettivi operativi. Questi potrebbero essere la soddisfazione del cliente, il tempo trascorso dall'ideazione della funzionalità alla sua implementazione, il tempo medio di risoluzione dei problemi e altro.

Partendo dai KPI, identifica le metriche e le origini di dati che potrebbero rispecchiare al meglio questi obiettivi. La soddisfazione del cliente può essere una combinazione di diverse metriche, come i tempi di attesa o di risposta durante le chiamate, i punteggi di soddisfazione e i tipi di problemi sollevati. I tempi di implementazione possono essere la somma del tempo necessario per il test e l'implementazione, con l'aggiunta di eventuali correzioni post-implementazione. Le statistiche che mostrano il tempo dedicato a diversi tipi di problemi (o il numero di tali problemi) possono fornire indicazioni su dove è necessario un impegno mirato.

Risorse

Documenti correlati:

- Amazon QuickSight - Using KPIs
- Amazon CloudWatch - Using Metrics
- Creazione di pannelli di controllo
- How to track your cost optimization KPIs with KPI Dashboard

OPS09-BP02 Comunicare lo stato e le tendenze per garantire la visibilità delle operazioni

Conoscere lo stato delle operazioni e la direzione verso la quale tendono a muoversi è necessario per identificare quando i risultati possono essere a rischio, se è possibile supportare o meno carichi di lavoro aggiuntivi o per verificare gli effetti che le modifiche hanno avuto sui team. Durante gli eventi operativi, disporre di pagine di stato a cui gli utenti e i team operativi possono fare riferimento per ottenere informazioni può ridurre la pressione sui canali di comunicazione e diffondere informazioni in modo proattivo.

Risultato desiderato:

- I responsabili delle operazioni hanno a disposizione informazioni dettagliate per conoscere il volume di chiamate che i loro team stanno gestendo e quali operazioni sono in corso, ad esempio le implementazioni.
- Quando si verificano eventi che possono compromettere le normali operazioni, vengono inviati avvisi agli stakeholder e alle comunità di utenti.
- Quando ricevono un avviso o si verifica un problema, la leadership dell'organizzazione e gli stakeholder possono controllare una pagina di stato e ottenere informazioni relative a un evento operativo, come punti di contatto, informazioni sui ticket e tempi di ripristino stimati.
- I report messi a disposizione della leadership e degli stakeholder contengono statistiche operative come il volume delle chiamate in un periodo di tempo, i punteggi di soddisfazione degli utenti, il numero e l'età di ticket in sospeso.

Anti-pattern comuni:

- Se un carico di lavoro si interrompe, il servizio diventa non disponibile. Il volume delle chiamate aumenta quando gli utenti chiedono di sapere cosa sta succedendo. Le richieste dei manager di sapere chi sta risolvendo un problema comportano un ulteriore aumento del volume. Vari team operativi duplicano gli sforzi mentre effettuano indagini.
- La volontà di acquisire una nuova capacità porta a riassegnare gli sforzi di alcuni membri del personale verso compiti di tipo tecnico. Non viene fornito alcun backfill e i tempi di risoluzione

dei problemi aumentano. Queste informazioni non vengono acquisite e i manager vengono a conoscenza del problema solo dopo diverse settimane o quando viene ricevuto il feedback negativo degli utenti.

Vantaggi dell'adozione di questa best practice: A volte, durante eventi operativi che hanno un impatto sull'azienda, si spreca molto tempo ed energia in query per ottenere informazioni da vari team nel tentativo di comprendere la situazione. Grazie alla creazione di pagine di stato e dashboard ampiamente diffuse, gli stakeholder possono ottenere rapidamente informazioni, ad esempio, se è stato rilevato o meno un problema, chi è a capo delle attività di risoluzione o quando è previsto un ritorno alle normali operazioni. Ciò permette ai membri del team di avere più tempo per affrontare i problemi, perché non devono dilungarsi a comunicare lo stato agli altri.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Crea dashboard che mostrino le metriche fondamentali attuali per i tuoi team operativi e rendile facilmente accessibili ai responsabili operativi e ai manager.

Crea pagine di stato aggiornabili rapidamente per diffondere informazioni relative a un incidente o un evento, come chi ne è responsabile e chi coordina la risposta. Condividi in questa pagina eventuali passaggi o soluzioni alternative che gli utenti dovrebbero prendere in considerazione e divulga ampiamente la posizione della pagina. Incoraggia gli utenti a controllare prima questa pagina quando si trovano di fronte a un problema sconosciuto.

Raccogli e fornisci report che mostrino lo stato di salute delle operazioni nel tempo e distribuiscili a leader e responsabili decisionali per illustrare il lavoro dei team operativi e le loro sfide ed esigenze.

Condividi con i team le metriche e i report che meglio riflettono gli obiettivi e i KPI e come hanno influito nel guidare il cambiamento. Dedica del tempo a queste attività per aumentare l'importanza delle operazioni nei e tra i team.

Risorse

Documenti correlati:

- Measure Progress
- Creazione di pannelli di controllo per visibilità operativa

Soluzioni correlate:

- Data Operations

OPS09-BP03 Revisione delle metriche operative e assegnazione delle priorità per favorire il miglioramento

L'assegnazione di tempo e risorse per la revisione dello stato delle operazioni garantisce che servire il settore d'attività rimanga una priorità quotidiana. Effettua regolarmente riunioni con i responsabili operativi e gli stakeholder per rivedere le metriche, riconfermare o modificare traguardi e obiettivi e dare priorità ai miglioramenti.

Risultato desiderato:

- I responsabili operativi e il personale si incontrano regolarmente per esaminare le metriche in un determinato periodo di riferimento. Si comunicano le sfide, si celebrano le vittorie e si condividono le lezioni apprese.
- Gli stakeholder e i leader aziendali vengono regolarmente informati sullo stato delle operazioni e sollecitati a fornire input su obiettivi, KPI e iniziative future. Vengono discusse e contestualizzate le scelte tra erogazione dei servizi, operazioni e manutenzione.

Anti-pattern comuni:

- Viene lanciato un nuovo prodotto, ma i team operativi di livello 1 e 2 non sono adeguatamente formati per fornire supporto oppure non dispongono di personale aggiuntivo. I leader non vedono le metriche che mostrano la diminuzione dei tempi di risoluzione dei ticket e l'aumento del volume degli incidenti. Si agisce settimane dopo, quando i numeri delle sottoscrizioni iniziano a diminuire a causa di utenti scontenti che abbandonano la piattaforma.
- Da molto tempo esiste un processo manuale per eseguire la manutenzione su un carico di lavoro. La volontà di automatizzare, seppur presente, costituiva una priorità bassa data la scarsa importanza del sistema. Nel corso del tempo, tuttavia, l'importanza del sistema è cresciuta e ora i team operativi sono impegnati per la maggior parte del tempo in questi processi manuali. Non sono previste risorse per fornire una maggiore strumentazione ai team operativi oberati dall'aumento dei carichi di lavoro, con rischi di burnout per il personale. La leadership viene a conoscenza del problema una volta che viene segnalato da un membro del personale che lascia l'azienda per un concorrente.

Vantaggi dell'adozione di questa best practice: In alcune organizzazioni, può diventare difficile dedicare lo stesso tempo e la stessa attenzione alla fornitura di servizi e a nuovi prodotti od offerte.

Quando ciò si verifica, il settore d'attività può risentirne a causa del lento deterioramento del livello di servizio atteso. Questo perché le operazioni non cambiano e non si evolvono di pari passo con la crescita del business e possono diventare presto obsolete. Senza una revisione regolare delle informazioni raccolte dai team operativi, il rischio che l'azienda corre potrebbe diventare visibile solo quando è troppo tardi. Dedicare tempo alla revisione delle metriche e delle procedure insieme al personale operativo e alla leadership, permette di mettere in luce il ruolo cruciale svolto dai team operativi nell'identificare i rischi molto prima che raggiungano livelli critici. I team operativi ottengono una visione migliore dei cambiamenti e delle iniziative aziendali imminenti, il che permette di intraprendere azioni proattive. Grazie alla visibilità delle metriche operative, la leadership è consapevole del ruolo che i team operativi svolgono nel garantire la soddisfazione dei clienti, sia interni che esterni, ed è in grado di valutare meglio le scelte in base alle priorità o di garantire che ci sia sufficiente tempo per modificare e fare evolvere operazioni e risorse attraverso nuove iniziative aziendali e di carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Dedica del tempo alla revisione delle metriche operative con gli stakeholder e i team operativi e alla revisione dei dati dei report. Inserisci questi report nel contesto degli scopi e degli obiettivi dell'organizzazione per stabilire se vengono raggiunti. Identifica le cause di ambiguità quando gli obiettivi non sono chiari o possono esserci conflitti tra ciò che viene chiesto e ciò che viene fornito.

Identifica come il tempo, le persone e gli strumenti possono contribuire agli esiti delle operazioni. Stabilisci quali KPI ne verrebbero influenzati e quali devono essere gli obiettivi di successo. Effettua regolarmente una revisione per assicurarti che i team operativi dispongano di risorse sufficienti per supportare il settore d'attività.

Risorse

Documenti correlati:

- Amazon Athena
- Documentazione di riferimento su parametri e dimensioni di Amazon CloudWatch
- Amazon QuickSight
- AWS Glue
- AWS Glue Data Catalog
- Collect metrics and logs from Amazon EC2 instances and on-premises servers with the Amazon CloudWatch Agent

- Utilizzare i parametri Amazon CloudWatch

OPS 10. In che modo gestisci gli eventi del carico di lavoro e delle operazioni?

Prepara e convalida le procedure in risposta agli eventi per ridurre al minimo il loro impatto sul tuo carico di lavoro.

Best practice

- OPS10-BP01 Utilizzo di un processo per la gestione di eventi, incidenti e problemi
- OPS10-BP02 Definizione di un processo per ogni avviso
- OPS10-BP03 Definizione della priorità degli eventi operativi in base agli effetti sul business
- OPS10-BP04 Definizione dei percorsi di escalation
- OPS10-BP05 Definizione di un piano di comunicazione con i clienti per eventi che incidono sul servizio
- OPS10-BP06 Comunicazione dello stato tramite pannelli di controllo
- OPS10-BP07 Automazione delle risposte agli eventi

OPS10-BP01 Utilizzo di un processo per la gestione di eventi, incidenti e problemi

La capacità di gestire in modo efficiente eventi, incidenti e problemi è fondamentale per mantenere l'integrità e le prestazioni del carico di lavoro. È essenziale riconoscere e comprendere le differenze tra questi elementi per sviluppare una strategia di risposta e risoluzione efficace. Stabilire e seguire un processo ben definito per ogni aspetto facilita la gestione rapida ed efficace da parte del tuo team di qualsiasi sfida operativa che si presenti.

Risultato desiderato: La tua organizzazione gestisce efficacemente eventi operativi, incidenti e problemi attraverso processi ben documentati e archiviati a livello centrale. Questi processi vengono costantemente aggiornati per riflettere le modifiche, semplificando la gestione e mantenendo l'affidabilità del servizio e delle prestazioni dei carichi di lavoro elevata.

Anti-pattern comuni:

- Rispondi in modo reattivo, anziché proattivo, agli eventi.
- Vengono adottati approcci incoerenti a diversi tipi di eventi o incidenti.
- La tua organizzazione non effettua analisi e non impara dagli incidenti per prevenire eventi futuri.

Vantaggi dell'adozione di questa best practice:

- Processi di risposta semplificati e standardizzati.
- Riduzione dell'impatto degli incidenti su servizi e clienti.
- Risoluzione rapida dei problemi.
- Miglioramento continuo dei processi operativi.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

L'implementazione di questa best practice prevede la registrazione degli eventi dei carichi di lavoro. Per la gestione di incidenti e problemi, è necessario ricorrere ai processi. I processi sono documentati, condivisi e aggiornati con frequenza. I problemi vengono identificati, classificati in base alla priorità e corretti.

Comprensione di eventi, incidenti e problemi

- Eventi: Un evento può essere l'adempimento di un'azione, un'occorrenza o un cambiamento di stato. Gli eventi possono essere pianificati o non pianificati e possono avere origine all'interno o all'esterno del carico di lavoro.
- Incidenti: Gli incidenti sono eventi che richiedono una risposta, come interruzioni non pianificate o il peggioramento della qualità del servizio. Rappresentano interruzioni che richiedono un'attenzione immediata al fine di ripristinare il normale funzionamento del carico di lavoro.
- Problemi: I problemi sono le cause alla base di uno o più incidenti. Identificare e risolvere i problemi implica approfondire gli incidenti per prevenire eventi futuri.

Passaggi dell'implementazione

Eventi

1. Monitora gli eventi:
   - Implementa l'osservabilità e utilizza l'osservabilità del carico di lavoro.
   - Le azioni di monitoraggio intraprese da un utente, ruolo o servizio AWS vengono registrate come eventi in AWS CloudTrail.
   - Rispondi alle modifiche operative delle tue applicazioni in tempo reale con Amazon EventBridge.

- Valuta, monitora e registra continuamente le modifiche alla configurazione delle risorse con AWS Config.

2. Crea processi:
   - Sviluppa un processo per valutare quali eventi sono significativi e richiedono di essere monitorati. Ciò comporta l'impostazione di soglie e parametri per le attività normali e anomale.
   - Determina i criteri in base ai quali un evento viene segnalato come un incidente. Ad esempio, la gravità dell'evento, l'impatto sugli utenti o la deviazione dal comportamento previsto.
   - Rivedi regolarmente i processi di monitoraggio e risposta agli eventi. Ciò include l'analisi degli incidenti passati, l'adeguamento delle soglie e il perfezionamento dei meccanismi di avviso.

Incidenti

1. Rispondi agli incidenti:
   - Usa gli approfondimenti degli strumenti di osservabilità per identificare e rispondere rapidamente agli incidenti.
   - Implementa OpsCenter di AWS Systems Manager per aggregare, organizzare e dare priorità agli elementi operativi e agli incidenti.
   - Utilizza servizi come Amazon CloudWatch e AWS X-Ray per un'analisi e una risoluzione dei problemi più approfondite.
   - Considera il servizio AWS Managed Services (AMS) per una migliore gestione degli incidenti, grazie alle sue capacità proattive, preventive e investigative. AMS estende il supporto operativo con servizi come monitoraggio, rilevamento e risposta agli incidenti e gestione della sicurezza.
   - Per i clienti del supporto Enterprise è disponibile Rilevamento e risposta agli incidenti di AWS, che fornisce il monitoraggio proattivo continuo e la gestione degli incidenti per i carichi di lavoro di produzione.
2. Crea un processo di gestione degli incidenti:
   - Definisci un processo strutturato di gestione degli incidenti, che includa ruoli, protocolli di comunicazione e passaggi per la risoluzione chiari.
   - Integra la gestione degli incidenti con strumenti come AWS Chatbot per una risposta e un coordinamento efficienti.
   - Classifica gli incidenti in base alla gravità, con piani di risposta agli incidenti predefiniti per ogni categoria.
3. Apprendi e migliora:

- Conduci analisi post-incidente per comprendere le cause principali e trovare una risoluzione efficace.
- Aggiorna e migliora continuamente i piani di risposta in base alle revisioni e alle pratiche in evoluzione.
- Documenta e condividi le lezioni apprese tra i team per migliorare la resilienza operativa.
- I clienti del supporto Enterprise possono richiedere di seguire il workshop relativo alla gestione degli incidenti al proprio Technical Account Manager (TAM). Questo workshop guidato consente di verificare il piano di risposta agli incidenti esistente e ti aiuta a individuare eventuali aree da migliorare.

Problemi

1. Identifica i problemi:
   - Utilizza i dati degli incidenti passati per identificare modelli ricorrenti che potrebbero indicare la presenza di problemi sistemici più profondi.
   - Usa strumenti come AWS CloudTrail e Amazon CloudWatch per analizzare le tendenze e scoprire i problemi sottostanti.
   - Coinvolgi team interfunzionali, ad esempio i team dediti alle operazioni, allo sviluppo e i reparti aziendali, per ottenere prospettive diverse sulle cause principali.
2. Crea un processo di gestione dei problemi:
   - Sviluppa un processo strutturato per la gestione dei problemi, concentrandoti su soluzioni a lungo termine piuttosto che su correzioni rapide.
   - Incorpora tecniche di analisi delle cause principali (RCA) per indagare e comprendere le cause alla base degli incidenti.
   - Aggiorna le policy e le procedure operative e l'infrastruttura in base ai risultati per prevenire il ripetersi degli incidenti.
3. Continua a migliorare:
   - Promuovi una cultura di apprendimento e miglioramento continui, incoraggiando i team a identificare e affrontare in modo proattivo i problemi potenziali.
   - Analizza e rivedi regolarmente i processi e gli strumenti di gestione dei problemi per allinearli agli scenari aziendali e tecnologici in evoluzione.
   - Condividi approfondimenti e best practice in tutta l'organizzazione per creare un ambiente operativo più resiliente ed efficiente.

4. Integra AWS Support:
    - Utilizza risorse di supporto AWS, come AWS Trusted Advisor, per una guida proattiva e suggerimenti in merito all'ottimizzazione.
    - I clienti del supporto Enterprise possono accedere a programmi specializzati come Countdown AWS per ricevere supporto durante gli eventi critici.
    -

Livello di impegno per il piano di implementazione: Medio

Risorse

Best practice correlate:

- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS04-BP02 Implementazione della telemetria dell'applicazione
- OPS07-BP03 Utilizzo di runbook per eseguire le procedure
- OPS07-BP04 Utilizzo dei playbook per analizzare i problemi
- OPS08-BP01 Analisi delle metriche del carico di lavoro
- OPS11-BP02 Esecuzione di analisi post-incidente

Documenti correlati:

- AWS Security Incident Response Guide
- Rilevamento e risposta agli incidenti di AWS
- AWS Cloud Adoption Framework: Operations Perspective - Incident and problem management
- Incident Management in the Age of DevOps and SRE (Gestione degli incidenti nell'era di DevOps e SRE)
- PagerDuty - What is Incident Management? (PagerDuty - Che cos'è la gestione degli incidenti?)

Video correlati:

- Top incident response tips from AWS
- AWS re:Invent 2022 - The Amazon Builders' Library: 25 yrs of Amazon operational excellence
- AWS re:Invent 2022 - AWS Incident Detection and Response (SUP201)

- Introducing Incident Manager from AWS Systems Manager

Esempi correlati:

- AWS Proactive Services – Incident Management Workshop
- How to Automate Incident Response with PagerDuty and AWS Systems Manager Incident Manager
- Engage Incident Responders with the On-Call Schedules in AWS Systems Manager Incident Manager
- Improve the Visibility and Collaboration during Incident Handling in AWS Systems Manager Incident Manager
- Incident reports and service requests in AMS

Servizi correlati:

- Amazon EventBridge

OPS10-BP02 Definizione di un processo per ogni avviso

Stabilire un processo chiaro e definito per ogni avviso nel sistema è essenziale per una gestione efficace ed efficiente degli incidenti. Questa pratica garantisce che ogni avviso porti a una risposta specifica e attuabile, migliorando l'affidabilità e la reattività delle operazioni.

Risultato desiderato: Ogni avviso avvia un piano di risposta specifico e ben definito. Ove possibile, le risposte sono automatizzate e dotate di una chiara titolarità e di un percorso di escalation definito. Gli avvisi sono collegati a una base di conoscenze aggiornata, in modo che qualsiasi operatore sia in grado di rispondere in modo coerente ed efficace. Le risposte sono rapide e uniformi su tutta la linea, migliorando l'efficienza e l'affidabilità operativa.

Anti-pattern comuni:

- Gli avvisi non hanno un processo di risposta predefinito, il che porta a risoluzioni improvvisate e tardive.
- Il sovraccarico di avvisi comporta che gli avvisi importanti vengano trascurati.
- Gli avvisi vengono gestiti in modo incoerente a causa della mancanza di titolarità e responsabilità chiare.

Vantaggi dell'adozione di questa best practice:

- Vengono generati solo avvisi utilizzabili, il che riduce l'affaticamento da avvisi.
- Riduzione del tempo medio di risoluzione (MTTR) per problemi operativi.
- Riduzione del tempo medio di indagine (MTTI), il che aiuta a ridurre l'MTTR.
- Migliore capacità di scalare le risposte operative.
- Maggiore coerenza e affidabilità nella gestione degli eventi operativi.

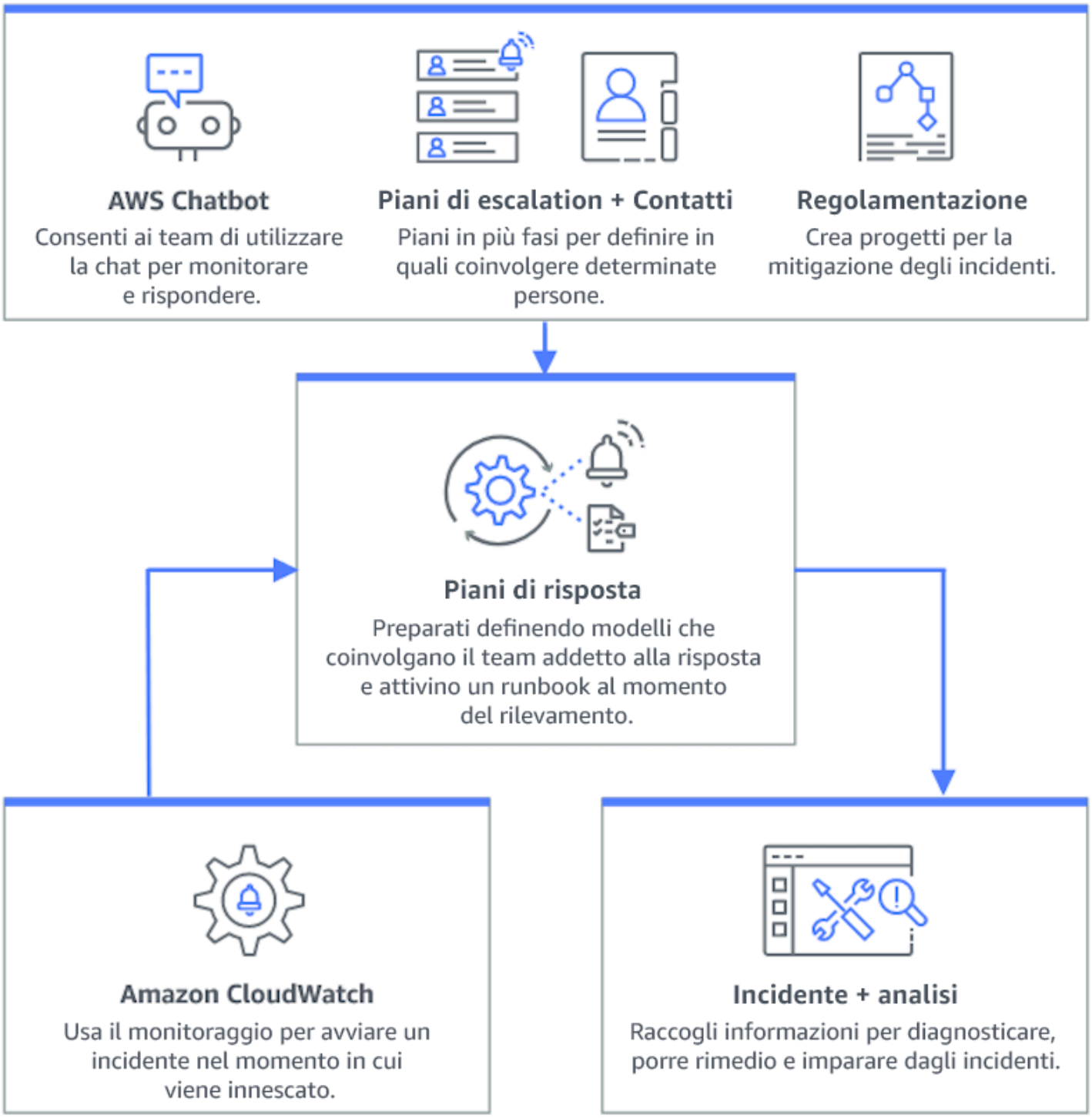
Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Avere un processo per ogni avviso implica stabilire un piano di risposta chiaro per ciascun avviso, automatizzare le risposte ove possibile e perfezionare continuamente questi processi in base al feedback operativo e all'evoluzione dei requisiti.

Passaggi dell'implementazione

Il diagramma seguente illustra il flusso di lavoro di gestione degli incidenti all'interno di AWS Systems Manager Incident Manager. È progettato per rispondere rapidamente ai problemi operativi creando automaticamente incidenti in risposta a eventi specifici che si verificano in Amazon CloudWatch oppure Amazon EventBridge. Quando viene creato automaticamente o manualmente un incidente, Incident Manager centralizza la gestione dell'incidente, organizza le informazioni pertinenti sulle risorse AWS e avvia piani di risposta predefiniti. Ciò include l'esecuzione dei runbook di Automazione Systems Manager per un'azione immediata e la creazione di un elemento di lavoro operativo principale in OpsCenter per tenere traccia delle attività e delle analisi correlate. Questo processo semplificato accelera e coordina la risposta agli incidenti in tutto l'ambiente AWS.



1. Utilizzo di allarmi compositi: Crea allarmi compositi in CloudWatch per raggruppare gli allarmi correlati, riducendo il rumore e consentendo risposte più significative.
2. Integra gli allarmi Amazon CloudWatch con Incident Manager Configura gli allarmi CloudWatch per creare automaticamente incidenti in AWS Systems Manager Incident Manager.

3. Integra Amazon EventBridge con Incident Manager: Crea regole EventBridge per reagire agli eventi e creare incidenti utilizzando piani di risposta definiti.
4. Preparazione per gli incidenti in Incident Manager:
    - Definisci piani di risposta dettagliati in Incident Manager per ogni tipo di avviso.
    - Stabilisci canali di chat tramite AWS Chatbot collegato ai piani di risposta in Incident Manager, facilitando la comunicazione in tempo reale durante gli incidenti su piattaforme come Slack, Microsoft Teams e Amazon Chime.
    - Incorpora i runbook di Automazione Systems Manager all'interno di Incident Manager per guidare le risposte automatiche agli incidenti.

Risorse

Best practice correlate:

- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS08-BP04 Creare avvisi fruibili

Documenti correlati:

- AWS Cloud Adoption Framework: Operations Perspective - Incident and problem management
- Utilizzo degli allarmi di Amazon CloudWatch
- Configurazione di AWS Systems Manager Incident Manager
- Preparazione per gli incidenti in Incident Manager

Video correlati:

- Top incident response tips from AWS

Esempi correlati:

- AWS Workshop - AWS Systems Manager Incident Manager - Automate incident response to security events

OPS10-BP03 Definizione della priorità degli eventi operativi in base agli effetti sul business

Rispondere tempestivamente agli eventi operativi è fondamentale, ma non tutti gli eventi sono uguali. Quando si assegnano le priorità in base all'impatto aziendale, si dà la priorità anche alla risoluzione di eventi che possono avere conseguenze significative, come la compromissione della sicurezza, perdite finanziarie, violazioni normative o danni alla reputazione.

Risultato desiderato: La priorità delle risposte agli eventi operativi si basa sul potenziale impatto dell'evento sulle operazioni e sugli obiettivi aziendali. Ciò rende le risposte efficienti ed efficaci.

Anti-pattern comuni:

- Ogni evento viene trattato con lo stesso livello di urgenza, generando confusione e ritardi nell'affrontare le criticità.
- Non è possibile distinguere tra eventi ad alto e basso impatto, con conseguente errata allocazione delle risorse.
- L'organizzazione non dispone di un chiaro framework di assegnazione delle priorità, il che genera risposte incoerenti agli eventi operativi.
- Agli eventi viene assegnata la priorità in base all'ordine in cui vengono segnalati piuttosto che al loro impatto sui risultati aziendali.

Vantaggi dell'adozione di questa best practice:

- Assicura che la risposta si concentri in primo luogo sulle funzioni aziendali critiche , riducendo al minimo i danni potenziali.
- Migliora l'allocazione delle risorse durante più eventi simultanei.
- Migliora la capacità dell'organizzazione di mantenere la fiducia e soddisfare i requisiti normativi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Di fronte a molteplici eventi operativi, è essenziale un approccio strutturato alla definizione delle priorità basato sull'impatto e sull'urgenza. Questo approccio consente di prendere decisioni informate, indirizzare gli sforzi laddove sono più necessari e mitigare il rischio per la continuità aziendale.

Passaggi dell'implementazione

1. Valuta l'impatto: Sviluppa un sistema di classificazione per valutare la gravità degli eventi in termini di potenziale impatto sulle operazioni e sugli obiettivi aziendali. L'esempio seguente mostra le categorie di impatto:

Livello di impatto	Descrizione
alto	Coinvolge molti dipendenti o clienti, ha un elevato impatto finanziario, genera un elevato danno alla reputazione o lesioni.
medio	Coinvolge un gruppo di dipendenti o clienti, ha un impatto finanziario moderato o genera un danno alla reputazione moderato.
basso	Coinvolge singoli dipendenti o clienti, ha un basso impatto finanziario o genera un danno alla reputazione di lieve entità.

2. Valuta l'urgenza: Definisci i livelli di urgenza in base alla rapidità con cui un evento deve ricevere una risposta, considerando fattori come la sicurezza, le implicazioni finanziarie e gli accordi sui livelli di servizio (SLA). L'esempio seguente illustra le categorie di urgenza:

Livello di urgenza	Descrizione
alto	Produce danni che aumentano in maniera esponenziale, incide su un lavoro sensibile al fattore tempo, escalation imminente, interessa utenti o gruppi VIP.
medio	Produce danni che aumentano nel tempo oppure interessa un singolo utente o gruppo VIP.
basso	Produce danni marginali che aumentano nel tempo o incide su lavori non sensibili al fattore tempo.

3. Crea una matrice di prioritizzazione:
    - Usa una matrice per incrociare impatto e urgenza, assegnando livelli di priorità a diverse combinazioni.
    - Rendi la matrice accessibile e comprensibile da tutti i membri del team responsabili delle risposte agli eventi operativi.
    - La seguente matrice di esempio mostra la gravità dell'incidente in base all'urgenza e all'impatto:

Urgenza e impatto	alto	Media	basso
alto	Critica	Urgente	alto
medio	Urgente	alto	Normale
basso	alto	Normale	basso

4. Forma e comunica: Forma i team di risposta sulla matrice di prioritizzazione e sull'importanza di attenersi ad essa durante un evento. Comunica il processo di definizione delle priorità a tutte le parti interessate per stabilire aspettative chiare.
5. Integrazione con la risposta agli incidenti:
    - Incorpora la matrice di prioritizzazione nei tuoi piani e strumenti di risposta agli incidenti.
    - Automatizza la classificazione e la prioritizzazione degli eventi, ove possibile, per accelerare i tempi di risposta.
    - I clienti del supporto Enterprise hanno a disposizione Rilevamento e risposta agli incidenti di AWS, che fornisce il monitoraggio proattivo 24 ore su 24, 7 giorni su 7 e la gestione degli incidenti per i carichi di lavoro di produzione.
6. Rivedi e adatta: Rivedi regolarmente l'efficacia del processo di definizione delle priorità e apporta modifiche in base al feedback e ai cambiamenti nell'ambiente aziendale.

Risorse

Best practice correlate:

- OPS03-BP03 Incoraggiamento all'escalation
- OPS08-BP04 Creare avvisi fruibili
- OPS09-BP01 Misura gli obiettivi operativi e i KPI con le metriche

Documenti correlati:

- Atlassian - Understanding incident severity levels
- IT Process Map - Checklist Incident Priority

OPS10-BP04 Definizione dei percorsi di escalation

Stabilisci percorsi di escalation chiari all'interno dei tuoi protocolli di risposta agli incidenti per facilitare un'azione tempestiva ed efficace. Ciò include la specificazione delle richieste relative all'escalation, la descrizione dettagliata del processo di escalation e la preapprovazione delle azioni per accelerare il processo decisionale e ridurre il tempo medio di risoluzione (MTTR).

Risultato desiderato: Un processo strutturato ed efficiente che inoltra gli incidenti al personale appropriato, riducendo al minimo i tempi di risposta e l'impatto.

Anti-pattern comuni:

- La mancanza di chiarezza in merito alle procedure di ripristino genera risposte improvvisate in caso di incidenti critici.
- L'assenza di autorizzazioni e titolarità definite comporta ritardi quando è necessaria un'azione urgente.
- Le parti interessate e i clienti non sono informati nei tempi attesi.
- Le decisioni importanti subiscono ritardi.

Vantaggi dell'adozione di questa best practice:

- Risposta semplificata agli incidenti tramite procedure di escalation predefinite.
- Tempi di inattività ridotti con azioni preapprovate e titolarità chiara.
- Migliore allocazione delle risorse e adeguamenti del livello di supporto in base alla gravità degli incidenti.
- Migliore comunicazione con le parti interessate e i clienti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I percorsi di escalation correttamente definiti sono fondamentali per una risposta rapida agli incidenti. AWS Systems Manager Incident Manager supporta l'impostazione di piani di escalation strutturati e di pianificazioni della reperibilità, che avvisano il personale pertinente preparandolo ad agire in caso di incidenti.

Passaggi dell'implementazione

1. Configura le richieste di escalation: Configura avvisi CloudWatch per creare un incidente in AWS Systems Manager Incident Manager.
2. Imposta la pianificazione della reperibilità: Crea la pianificazione della reperibilità in Incident Manager affinché sia in linea con i tuoi percorsi di escalation. Fornisci al personale di turno le autorizzazioni e gli strumenti necessari per agire rapidamente.
3. Procedure di escalation dettagliate:
   - Determina le condizioni specifiche in base alle quali un incidente deve essere inoltrato.
   - Crea piani di escalation in Incident Manager.
   - I canali di escalation devono consistere in un contatto o in una pianificazione della reperibilità.
   - Definisci i ruoli e le responsabilità del team a ogni livello di escalation.
4. Approva preventivamente le azioni di mitigazione: Collabora con i decision maker per approvare preventivamente le azioni per gli scenari previsti. Utilizza i runbook di Automazione Systems Manager integrati con Incident Manager per accelerare la risoluzione degli incidenti.
5. Specifica la proprietà: Identifica chiaramente i proprietari interni per ogni fase del percorso di escalation.
6. Fornisci dettagli in merito alle escalation a terze parti:
   - Documenta gli accordi sui livelli di servizio (SLA) di terze parti e adeguali agli obiettivi interni.
   - Stabilisci protocolli chiari per la comunicazione con i fornitori durante gli incidenti.
   - Integra i contatti dei fornitori negli strumenti di gestione degli incidenti per l'accesso diretto.
   - Conduci regolarmente esercitazioni che includano scenari di risposta di terze parti.
   - Mantieni le informazioni sulle escalation dei fornitori ben documentate e facilmente accessibili.
7. Esegui formazione e test per i piani di escalation: Forma il tuo team sul processo di escalation e conduci regolarmente esercitazioni di risposta agli incidenti o game day. I clienti del supporto Enterprise possono richiedere di seguire un workshop relativo alla gestione degli incidenti.
8. Continua a migliorare: Verifica regolarmente l'efficacia dei tuoi percorsi di escalation. Aggiorna i tuoi processi in base alle lezioni apprese dalle analisi degli incidenti e dal feedback continuo.

Livello di impegno per il piano di implementazione: Moderato

Risorse

Best practice correlate:

- OPS08-BP04 Creare avvisi fruibili
- OPS10-BP02 Definizione di un processo per ogni avviso
- OPS11-BP02 Esecuzione di analisi post-incidente

Documenti correlati:

- Piani di escalation di AWS Systems Manager Incident Manager
- Working with on-call schedules in Incident Manager
- Creating and Managing Runbooks
- Temporary elevated access management with AWS IAM Identity Center
- Atlassian - Escalation policies for effective incident management

OPS10-BP05 Definizione di un piano di comunicazione con i clienti per eventi che incidono sul servizio

Una comunicazione efficace durante gli eventi che incidono sul servizio è fondamentale per mantenere la fiducia e la trasparenza con i clienti. Un piano di comunicazione ben definito sostiene la comunicazione rapida e chiara di informazioni all'interno e all'esterno dell'organizzazione durante gli incidenti.

Risultato desiderato:

- Un solido piano di comunicazione che informa efficacemente i clienti e le parti interessate durante gli eventi che influiscono sul servizio.
- Trasparenza nella comunicazione per creare fiducia e ridurre la preoccupazione dei clienti.
- Riduzione al minimo dell'impatto che gli eventi che incidono sul servizio hanno sull'esperienza del cliente e sulle operazioni aziendali.

Anti-pattern comuni:

- Una comunicazione inadeguata o in ritardo genera confusione e insoddisfazione nei clienti.

- Una messaggistica eccessivamente tecnica o vaga impedisce la comunicazione dell'impatto effettivo sugli utenti.
- È assente una strategia di comunicazione predefinita, con conseguente messaggistica incoerente e reattiva.

Vantaggi dell'adozione di questa best practice:

- Maggiore fiducia e soddisfazione dei clienti attraverso una comunicazione chiara e proattiva.
- Riduzione del carico operativo per i team di supporto grazie alla risoluzione preventiva delle preoccupazioni dei clienti.
- Maggiore efficienza di gestione e risoluzione degli incidenti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La creazione di un piano di comunicazione completo per gli eventi che incidono sul servizio implica prendere in considerazione molteplici aspetti, dalla scelta dei canali giusti alla creazione del messaggio e del tono. Il piano deve essere adattabile, scalabile e soddisfare diversi scenari di interruzione del servizio.

Passaggi dell'implementazione

1. Definisci ruoli e responsabilità:
    - Assegna a un responsabile degli incidenti gravi la supervisione delle attività di risposta agli incidenti.
    - Designa un responsabile delle comunicazioni dedicato al coordinamento di tutte le comunicazioni esterne e interne.
    - Includi il responsabile dell'assistenza per fornire una comunicazione coerente attraverso ticket di supporto.
2. Identifica i canali di comunicazione: Seleziona canali come chat aziendale, e-mail, SMS, social media, notifiche in-app e pagine di stato. Questi canali devono essere resilienti e in grado di operare in maniera indipendente durante gli eventi che incidono sul servizio.
3. Comunica in modo rapido, chiaro e regolare con i clienti:

- Sviluppa modelli per vari scenari di compromissione del servizio, focalizzandoti sulla semplicità e sui dettagli essenziali. Includi informazioni sul problema relativo al servizio, sui tempi di risoluzione previsti e sull'impatto.
- Usa Amazon Pinpoint per avvisare i clienti tramite notifiche push, notifiche in-app, e-mail, SMS, messaggi vocali e messaggi su canali personalizzati.
- Usa Amazon Simple Notification Service (Amazon SNS) per avvisare gli abbonati in modo programmatico o tramite e-mail, notifiche push su dispositivi mobili e SMS.
- Comunica lo stato tramite dashboard condividendo pubblicamente una dashboard Amazon CloudWatch.
- Incoraggia il coinvolgimento sui social media:
  - Monitora attivamente i social media per comprendere il sentimento dei clienti.
  - Pubblica post su piattaforme di social media per aggiornare il pubblico e coinvolgere la comunità.
  - Prepara modelli per una comunicazione coerente e chiara sui social media.

4. Coordina la comunicazione interna: Implementa protocolli interni utilizzando strumenti come AWS Chatbot per migliorare il coordinamento e la comunicazione tra i team. Usa le dashboard CloudWatch per comunicare lo stato.
5. Orchestra la comunicazione con strumenti e servizi dedicati:
   - Usa AWS Systems Manager Incident Manager con AWS Chatbot per configurare canali di chat dedicati per la comunicazione interna e il coordinamento in tempo reale durante gli incidenti.
   - Usa i runbook AWS Systems Manager Incident Manager per automatizzare le notifiche ai clienti durante gli incidenti tramite Amazon Pinpoint, Amazon SNS o strumenti di terze parti come le piattaforme di social media.
   - Incorpora i flussi di lavoro di approvazione all'interno dei runbook per rivedere e autorizzare tutte le comunicazioni esterne prima dell'invio.
6. Fai pratica e migliora:
   - Tieni corsi di formazione sull'uso di strumenti e strategie di comunicazione. Responsabilizza i team affinché siano in grado di prendere decisioni tempestive durante gli incidenti.
   - Testa il piano di comunicazione con esercitazioni regolari o game day. Usa questi test per perfezionare la messaggistica e valutare l'efficacia dei canali.
   - Implementa meccanismi di feedback per valutare l'efficacia della comunicazione durante gli incidenti. Sviluppa continuamente il piano di comunicazione in base al feedback e alle esigenze mutevoli.

Livello di impegno per il piano di implementazione: alto

Risorse

Best practice correlate:

- OPS07-BP03 Utilizzo di runbook per eseguire le procedure
- OPS10-BP06 Comunicazione dello stato tramite pannelli di controllo
- OPS11-BP02 Esecuzione di analisi post-incidente

Documenti correlati:

- Atlassian - Incident communication best practices
- Atlassian - How to write a good status update
- PagerDuty - A Guide to Incident Communications

Video correlati:

- Atlassian - Create your own incident communication plan: Incident templates

Esempi correlati:

- Dashboard AWS Health
- Esempi di aggiornamenti di stato AWS

OPS10-BP06 Comunicazione dello stato tramite pannelli di controllo

Usa le dashboard come strumento strategico per trasmettere lo stato operativo e le metriche fondamentali in tempo reale a diversi tipi di pubblico, inclusi team tecnici interni, leader e clienti. Queste dashboard offrono una rappresentazione visiva centralizzata dello stato del sistema e delle prestazioni aziendali, il che migliora la trasparenza e l'efficienza decisionale.

Risultato desiderato:

- Le dashboard forniscono una visione completa del sistema e delle metriche aziendali rilevanti per i diversi stakeholder.

- Le parti interessate possono accedere in modo proattivo alle informazioni operative, il che riduce la necessità di richieste di stato frequenti.
- Migliore processo decisionale in tempo reale durante le normali operazioni e gli incidenti.

Anti-pattern comuni:

- I tecnici che partecipano a una chiamata di gestione degli incidenti hanno bisogno di ricevere aggiornamenti di stato per poter agire rapidamente.
- Affidarsi alla reportistica manuale per la gestione comporta ritardi e potenziali imprecisioni.
- I team operativi vengono spesso interrotti per aggiornamenti sullo stato durante gli incidenti.

Vantaggi dell'adozione di questa best practice:

- Consente alle parti interessate di accedere immediatamente alle informazioni critiche, promuovendo un processo decisionale informato.
- Riduce le inefficienze operative riducendo al minimo i report manuali e le richieste di stato frequenti.
- Aumenta la trasparenza e la fiducia attraverso la visibilità in tempo reale delle prestazioni del sistema e delle metriche aziendali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Le dashboard comunicano efficacemente lo stato dei sistemi e le metriche aziendali e possono essere personalizzate in base alle esigenze di diversi gruppi di destinatari. Strumenti come le dashboard Amazon CloudWatch e Amazon QuickSight aiutano a creare dashboard interattive e in tempo reale per il monitoraggio del sistema e la business intelligence.

Passaggi dell'implementazione

1. Identifica le esigenze degli stakeholder: Determina le esigenze informative specifiche dei diversi gruppi di destinatari, come team tecnici, leader e clienti.
2. Scegli gli strumenti corretti: Seleziona strumenti appropriati come dashboard Amazon CloudWatch per il monitoraggio del sistema e Amazon QuickSight per una business intelligence interattiva.
3. Progetta dashboard efficaci:

- Progetta dashboard per presentare in modo chiaro metriche e KPI pertinenti, assicurandoti che siano comprensibili e utilizzabili.
- Incorpora visualizzazioni a livello di sistema e a livello aziendale, se necessario.
- Includi dashboard di alto livello (per ampie panoramiche) e di basso livello (per analisi dettagliate).
- Integra allarmi automatici all'interno di dashboard per evidenziare i problemi critici.
- Annota le dashboard con soglie e obiettivi delle metriche importanti per una visibilità immediata.

4. Integra le fonti di dati:
   - Utilizza Amazon CloudWatch per aggregare e visualizzare le metriche di vari servizi AWS e metriche di query da altre fonti di dati, creando una visione unificata delle metriche aziendali e dello stato del sistema.
   - Usa funzionalità come CloudWatch Logs Insights per effettuare query e visualizzare i dati di log provenienti da diverse applicazioni e servizi.
5. Fornisci l'accesso self-service:
   - Condividi le dashboard CloudWatch con le parti interessate per l'accesso self-service alle informazioni utilizzando funzionalità di condivisione della dashboard.
   - Assicurati che le dashboard siano facilmente accessibili e contengano informazioni aggiornate in tempo reale.
6. Aggiorna e perfeziona regolarmente:
   - Aggiorna e perfeziona continuamente le dashboard per allinearle alle esigenze aziendali in evoluzione e al feedback degli stakeholder.
   - Rivedi regolarmente le dashboard per assicurarti che siano sempre pertinenti ed efficaci nella trasmissione delle informazioni necessarie.

Risorse

Best practice correlate:

- OPS08-BP05 Creare dashboard

Documenti correlati:

- Creazione di dashboard per visibilità operativa
- Utilizzo dei pannelli di controllo Amazon CloudWatch

- Crea dashboard flessibili con variabili della dashboard
- Condivisione di dashboard CloudWatch
- Metriche di query da altre fonti di dati
- Aggiungi un widget personalizzato a una dashboard CloudWatch

Esempi correlati:

- One Observability Workshop - Dashboards

OPS10-BP07 Automazione delle risposte agli eventi

L'automazione delle risposte agli eventi è fondamentale per una gestione operativa rapida, coerente e priva di errori. Crea processi semplificati e utilizza strumenti per gestire e rispondere automaticamente agli eventi, riducendo al minimo gli interventi manuali e migliorando l'efficacia operativa.

Risultato desiderato:

- Riduzione degli errori umani e tempi di risoluzione più rapidi grazie all'automazione.
- Gestione degli eventi operativi coerente e affidabile.
- Maggiore efficienza operativa e affidabilità del sistema.

Anti-pattern comuni:

- La gestione manuale degli eventi comporta ritardi ed errori.
- L'automazione viene trascurata nelle attività ripetitive e critiche.
- Le attività manuali ripetitive causano affaticamento da avvisi e la mancata identificazione di problemi critici.

Vantaggi dell'adozione di questa best practice:

- Risposte agli eventi accelerate, riduzione dei tempi di inattività del sistema.
- Operazioni affidabili con gestione automatizzata e coerente degli eventi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Incorpora l'automazione per creare flussi di lavoro operativi efficienti e ridurre al minimo gli interventi manuali.

Passaggi dell'implementazione

1. Identifica le opportunità di automazione: Definisci le attività ripetitive da automatizzare, come la risoluzione dei problemi, l'arricchimento dei ticket, la gestione della capacità, la scalabilità, le implementazioni e i test.
2. Identifica i prompt di automazione:
   - Valuta e definisci condizioni o metriche specifiche al verificarsi delle quali inviare risposte automatiche utilizzando le azioni di allarme di Amazon CloudWatch.
   - Utilizza Amazon EventBridge per rispondere agli eventi nei servizi AWS, nei carichi di lavoro personalizzati e nelle applicazioni SaaS.
   - Prendi in considerazione eventi scatenanti come voci di log specifiche, soglie delle metriche delle prestazioni o cambiamenti di stato nelle risorse AWS.
3. Implementa l'automazione basata sugli eventi:
   - Usa i runbook di Automazione AWS Systems Manager per semplificare le attività di manutenzione, implementazione e bonifica.
   - Quando vengono creati incidenti in Incident Manager, i dettagli relativi alle risorse AWS coinvolte nell'incidente vengono raccolti e aggiunti automaticamente.
   - Monitora in modo proattivo le quote utilizzando Monitoraggio delle quote per AWS.
   - Regola automaticamente la capacità con AWS Auto Scaling per mantenere la disponibilità e le prestazioni.
   - Automatizza le pipeline di sviluppo con Amazon CodeCatalyst.
   - Smoke test o monitoraggio continuo di endpoint e API utilizzando il monitoraggio sintetico.
4. Esegui la mitigazione del rischio attraverso l'automazione:
   - Implementa risposte di sicurezza automatizzate per affrontare rapidamente i rischi.
   - Utilizza AWS Systems Manager State Manager per ridurre la deviazione delle configurazioni.
   - Correggi le risorse non conformi con Regole di AWS Config.

Livello di impegno per il piano di implementazione: alto

Risorse

Best practice correlate:

- OPS08-BP04 Creare avvisi fruibili
- OPS10-BP02 Definizione di un processo per ogni avviso

Documenti correlati:

- Using Systems Manager Automation runbooks with Incident Manager
- Creating incidents in Incident Manager
- AWS Service Quotas
- Monitor resource usage and send notifications when approaching quotas
- AWS Auto Scaling
- What is Amazon CodeCatalyst?
- Utilizzo degli allarmi di Amazon CloudWatch
- Utilizzo delle azioni di allarme di Amazon CloudWatch
- Remediating Noncompliant Resources with Regole di AWS Config
- Creazione di parametri da registro eventi mediante filtri
- AWS Systems Manager State Manager

Video correlati:

- Create Automation Runbooks with AWS Systems Manager
- How to automate IT Operations on AWS
- AWS Security Hub automation rules
- Start your software project fast with Amazon CodeCatalyst blueprints

Esempi correlati:

- Amazon CodeCatalyst Tutorial: Creating a project with the Modern three-tier web application blueprint
- One Observability Workshop
- Respond to incidents using Incident Manager

Evoluzione

Domanda

- OPS 11. In che modo fai evolvere le operazioni?

OPS 11. In che modo fai evolvere le operazioni?

Dedica tempo e risorse per ottenere un miglioramento incrementale pressoché continuo, per far evolvere l'efficacia e l'efficienza delle tue operazioni.

Best practice

- OPS11-BP01 Definizione di un processo per il miglioramento continuo
- OPS11-BP02 Esecuzione di analisi post-incidente
- OPS11-BP03 Implementazione di cicli di feedback
- OPS11-BP04 Gestione delle informazioni
- OPS11-BP05 Definizione dei fattori che promuovono il miglioramento
- OPS11-BP06 Convalida delle informazioni
- OPS11-BP07 Revisione dei parametri delle operazioni
- OPS11-BP08 Documentazione e condivisione delle conoscenze acquisite
- OPS11-BP09 Allocazione di tempo per i miglioramenti

OPS11-BP01 Definizione di un processo per il miglioramento continuo

Valuta il carico di lavoro rispetto alle best practice dell'architettura interna ed esterna. Effettua revisioni frequenti e deliberate del carico di lavoro. Dai priorità alle opportunità di miglioramento nella cadenza di sviluppo del software.

Risultato desiderato:

- Analizza di frequente il carico di lavoro rispetto alle best practice dell'architettura.
- Stabilisci per le opportunità di miglioramento la stessa priorità che assegni alle funzionalità del processo di sviluppo software.

Anti-pattern comuni:

- Non hai condotto una revisione dell'architettura del carico di lavoro da quando è stato implementato diversi anni fa.
- Stabilisci una priorità inferiore per le opportunità di miglioramento. Rispetto alle nuove funzionalità, queste opportunità rimangono nel backlog.
- Non esiste uno standard per l'implementazione delle modifiche alle best practice per l'organizzazione.

Vantaggi dell'adozione di questa best practice:

- Il carico di lavoro è aggiornato sulla base delle best practice di architettura.
- Fai evolvere il carico di lavoro in modo intenzionale.
- Puoi utilizzare le best practice dell'organizzazione per migliorare tutti i carichi di lavoro.
- Ottieni guadagni marginali che hanno un impatto cumulativo, con un incremento dell'efficienza.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Effettui di frequente la revisione dell'architettura del carico di lavoro. Utilizzi le best practice interne ed esterne per valutare il carico di lavoro e identificare le opportunità di miglioramento. Dai priorità alle opportunità di miglioramento nella cadenza di sviluppo del software.

Passaggi dell'implementazione

1. Esegui la revisione periodica dell'architettura del carico di lavoro di produzione secondo una frequenza concordata. Utilizza uno standard architettonico documentato che includa best practice specifiche di AWS.
   a. Usa gli standard definiti internamente per queste revisioni. Se non hai standard interni, usa Framework AWS Well-Architected.
   b. Utilizza AWS Well-Architected Tool per creare un obiettivo personalizzato delle best practice interne e condurre la revisione dell'architettura.
   c. Contatta un AWS Solution Architect o Technical Account Manager per condurre una revisione guidata di Framework Well-Architected del carico di lavoro.
2. Dai priorità alle opportunità di miglioramento identificate durante la revisione nel processo di sviluppo del software.

Livello di impegno per il piano di implementazione: Basso. Si può usare AWS Well-Architected Framework per eseguire la revisione annuale dell'architettura.

Risorse

Best practice correlate:

- OPS11-BP02 Esecuzione di analisi post-incidente
- OPS11-BP08 Documentazione e condivisione delle conoscenze acquisite
- OPS04 Implementazione dell'osservabilità

Documenti correlati:

- AWS Well-Architected Tool - Custom lenses
- AWS Well-Architected Whitepaper - The review process
- Customize Well-Architected Reviews using Custom Lenses and the AWS Well-Architected Tool
- Implementing the AWS Well-Architected Custom Lens lifecycle in your organization

Video correlati:

- Well-Architected Labs - Level 100: Custom Lenses on AWS Well-Architected Tool
- AWS re:Invent 2023 - Scaling AWS Well-Architected best practices across your organization

Esempi correlati:

- AWS Well-Architected Tool

OPS11-BP02 Esecuzione di analisi post-incidente

Esamina gli eventi che influiscono sui clienti e identifica i fattori che contribuiscono e le azioni preventive. Utilizza queste informazioni per sviluppare modi per limitare o prevenire il ripetersi degli incidenti. Sviluppa procedure per attivare risposte rapide ed efficaci. Comunica i fattori che hanno contribuito al presentarsi dell'imprevisto e le azioni correttive secondo necessità, specificamente mirate per il pubblico di destinazione.

Risultato desiderato:

- Stabilisci processi di gestione degli incidenti che includono l'analisi post-incidente.
- Hai a disposizione piani di osservabilità per raccogliere dati sugli eventi.
- Con questi dati comprendi e raccogli metriche che supportano il tuo processo di analisi post-incidente.
- Impari dagli incidenti per migliorare i risultati futuri.

Anti-pattern comuni:

- Sei amministratore di un server di applicazioni. Circa ogni 23 ore e 55 minuti tutte le sessioni attive vengono terminate. Hai tentato di identificare ciò che non va a buon fine sul server di applicazioni. Sospetti che potrebbe trattarsi di un problema di rete, ma non riesci a ottenere la collaborazione dal team di rete perché i suoi membri sono troppo occupati per supportarti. Ti manca un processo predefinito da seguire per ottenere supporto e raccogliere le informazioni necessarie per stabilire che cosa sta accadendo.
- Si è verificata una perdita di dati all'interno del carico di lavoro. Questa è la prima volta che si è verificata e la causa non è immediatamente identificabile. Decidi che non è importante perché puoi ricreare i dati. La perdita di dati inizia a verificarsi con maggiore frequenza e influisce sui clienti. Questo comporta inoltre un ulteriore onere operativo quando ripristini i dati mancanti.

Vantaggi dell'adozione di questa best practice:

- Disponendo di un processo predefinito per determinare i componenti, le condizioni, le azioni e gli eventi che hanno contribuito a un incidente, sei in grado di identificare le opportunità di miglioramento.
- Utilizzi i dati dell'analisi post-incidente per apportare miglioramenti.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Utilizza un processo per determinare i fattori che concorrenti. Esamina tutti gli incidenti che influiscono sul cliente. Predisponi un processo per identificare e documentare i fattori che contribuiscono a un incidente, in modo da sviluppare azioni di mitigazione in grado di limitare o impedire il suo ripetersi e per sviluppare procedure che consentano risposte rapide ed efficaci. Comunica le cause principali degli incidenti in modo appropriato e personalizza la comunicazione

in base al pubblico di destinazione. Condividi quanto appreso in maniera aperta all'interno della tua organizzazione.

Passaggi dell'implementazione

1. Raccogli metriche come le modifiche all'implementazione e alla configurazione, l'ora di inizio dell'incidente, l'ora dell'allarme, dell'intervento, dell'inizio della mitigazione e il tempo di risoluzione dell'incidente.
2. Descrivi i momenti fondamentali sulla linea temporale per comprendere gli eventi dell'incidente.
3. Poni le seguenti domande:
    a. Potresti migliorare il tempo di rilevamento?
    b. Sono presenti aggiornamenti alle metriche e agli allarmi che permettono di rilevare l'incidente prima?
    c. Puoi migliorare i tempi di diagnosi?
    d. Sono presenti aggiornamenti ai tuoi piani di risposta o di escalation che potrebbero coinvolgere prima i team di risposta corretti?
    e. Puoi migliorare il tempo necessario per la mitigazione?
    f. Ci sono passaggi del runbook o del playbook che potresti aggiungere o migliorare?
    g. È possibile prevenire che si verifichino incidenti futuri?
4. Crea liste di controllo e azioni. Monitora ed esegui tutte le azioni.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS11-BP01 Definizione di un processo per il miglioramento continuo
- OPS 4 - Implementazione dell'osservabilità

Documenti correlati:

- Performing a post-incident analysis in Incident Manager
- Operational Readiness Review

OPS11-BP03 Implementazione di cicli di feedback

I cicli di feedback forniscono informazioni fruibili che guidano il processo decisionale. Vanno creati nelle procedure e nei carichi di lavoro per identificare i problemi e le aree che necessitano di miglioramenti. Inoltre, convalidano gli investimenti effettuati nei miglioramenti. Questi cicli di feedback sono la base per migliorare continuamente il carico di lavoro.

I cicli di feedback si dividono in due categorie: feedback immediato e analisi retrospettiva. Il feedback immediato viene raccolto con la revisione delle prestazioni e dei risultati delle attività operative. Questo feedback proviene dai membri del team, dai clienti o dall'output automatizzato dell'attività. Il feedback immediato viene ricevuto ad esempio dal test A/B e dall'offerta di nuove funzionalità, ed è essenziale per anticipare l'errore (fail fast).

L'analisi retrospettiva viene eseguita regolarmente per acquisire il feedback della revisione dei risultati operativi e dei parametri nel tempo. Queste retrospettive si svolgono alla fine di uno sprint, in base a una cadenza o dopo importanti rilasci o eventi. Questo tipo di ciclo di feedback convalida gli investimenti nelle operazioni o nel carico di lavoro, consente di misurare il successo e comprova la tua strategia.

Risultato desiderato: l'uso del feedback immediato e dell'analisi retrospettiva per guidare i miglioramenti. L'applicazione di un meccanismo per acquisire il feedback di utenti e membri del team. L'uso dell'analisi retrospettiva per identificare le tendenze che guidano i miglioramenti.

Anti-pattern comuni:

- Lanci una nuova funzionalità ma non hai modo di ricevere il feedback dei clienti.
- Dopo aver investito in miglioramenti delle operazioni, non conduci una retrospettiva per convalidare gli investimenti.
- Raccogli il feedback dei clienti ma non lo esamini regolarmente.
- I cicli di feedback portano alla proposta di elementi di azione non sono inclusi nel processo di sviluppo software.
- I clienti non ricevono un feedback sui miglioramenti che hanno proposto.

Vantaggi dell'adozione di questa best practice:

- Puoi lavorare a ritroso con il cliente per promuovere nuove funzionalità.
- La cultura della tua organizzazione può reagire più rapidamente ai cambiamenti.
- Le tendenze vengono utilizzate per identificare le opportunità di miglioramento.

- Le retrospettive convalidano gli investimenti effettuati per il carico di lavoro e le operazioni.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

L'implementazione di questa best practice comporta l'utilizzo del feedback immediato e dell'analisi retrospettiva. Questi cicli di feedback guidano i miglioramenti. Esistono molti meccanismi per il feedback immediato, inclusi questionari, sondaggi dei clienti o moduli di feedback. La tua organizzazione utilizza anche le retrospettive per identificare le opportunità di miglioramento e convalidare le iniziative.

Esempio del cliente

AnyCompany Retail crea un modulo Web in cui i clienti possono fornire il feedback o segnalare problemi. Durante lo Scrum settimanale, il feedback degli utenti viene valutato dal team di sviluppo software. Il feedback viene regolarmente utilizzato per guidare l'evoluzione della piattaforma. Viene eseguita una retrospettiva alla fine di ogni sprint per identificare gli elementi che devono essere migliorati.

Passaggi dell'implementazione

1. Feedback immediato
   - Hai bisogno di un meccanismo per ricevere il feedback dai clienti e dai membri del team. Le attività operative possono anche essere configurate per fornire un feedback automatizzato.
   - L'organizzazione ha bisogno di un processo per rivedere il feedback, determinare cosa migliorare e pianificare il miglioramento.
   - Il feedback deve essere aggiunto al processo di sviluppo software.
   - Quando apporti miglioramenti, contatta l'autore del feedback.
     - Puoi utilizzare AWS Systems Manager OpsCenter per creare e monitorare questi miglioramenti come OpsItems.
2. Analisi retrospettiva
   - Conduci le retrospettive alla fine di un ciclo di sviluppo, a una cadenza prestabilita o dopo un rilascio importante.
   - Riunisci gli stakeholder coinvolti nel carico di lavoro per la riunione retrospettiva.
   - Crea tre colonne sulla lavagna o in un foglio di lavoro: Fine, Inizio e Mantenimento.
     - Fine è per tutto ciò che vuoi che il team smetta di fare.

    - Inizio è per le idee che vuoi iniziare ad applicare.
    - Mantenimento è per ciò che vuoi continuare a fare.
- Raccogli il feedback dagli stakeholder.
- Dai priorità al feedback. Assegna le azioni e gli stakeholder a qualsiasi elemento nelle colonne Inizio e Mantenimento.
- Aggiungi le azioni al processo di sviluppo software e comunica gli aggiornamenti sullo stato agli stakeholder mentre apporti i miglioramenti.

Livello di impegno per il piano di implementazione: medio. Per implementare questa best practice è necessario un modo per ricevere il feedback immediato e analizzarlo. Inoltre, è necessario stabilire un processo di analisi retrospettiva.

Risorse

Best practice correlate:

- OPS01-BP01 Valutazione delle esigenze dei clienti: i cicli di feedback sono un meccanismo per raccogliere le esigenze dei clienti esterni.
- OPS01-BP02 Valutazione delle esigenze dei clienti interni: gli stakeholder interni possono utilizzare i cicli di feedback per comunicare necessità e requisiti.
- OPS11-BP02 Esecuzione di analisi post-incidente: le analisi successive agli incidenti sono una forma importante di analisi retrospettiva da condurre dopo gli incidenti.
- OPS11-BP07 Revisione dei parametri delle operazioni: le revisioni dei parametri operativi identificano tendenze e aree di miglioramento.

Documenti correlati:

- 7 Pitfalls to Avoid When Building CCOE (7 errori da evitare durante la creazione di un Centro di eccellenza del Cloud (CCoE))
- Atlassian Team Playbook - Retrospectives (Playbook Atlassian Team - Retrospettive)
- Email Definitions: Feedback Loops (Definizioni di e-mail: cicli di feedback)
- Establishing Feedback Loops Based on the AWS Well-Architected Framework Review (Applicazione dei cicli di feedback in base alla revisione di Framework AWS Well-Architected)
- IBM Garage Methodology - Hold a retrospective (Metodologia IBM Garage - Condurre una retrospettiva)

- Investopedia - The PDCS Cycle (Investopedia - Il ciclo PDCA)
- Maximizing Developer Effectiveness by Tim Cochran (Massimizzazione dell'efficacia degli sviluppatori di Tim Cochran)
- Operations Readiness Reviews (ORR) Whitepaper - Iteration (Whitepaper per le revisioni della preparazione delle operazioni - Iterazione)
- TIL CSI - Continual Service Improvement (TIL CSI - Miglioramento continuo del servizio)
- When Toyota met e-commerce: Lean at Amazon (Toyota incontra l'e-commerce: semplificazione con Amazon)

Video correlati:

- Building Effective Customer Feedback Loops (Creazione di efficaci cicli di feedback dei clienti)

Esempi correlati:

- Astuto - Open source customer feedback tool (Astuto - Strumento di feedback dei clienti open source)
- AWS Solutions - QnABot on AWS (Soluzioni AWS - QnABot in AWS)
- Fider - A platform to organize customer feedback (Fider - Una piattaforma per organizzare il feedback dei clienti)

Servizi correlati:

- AWS Systems Manager OpsCenter

OPS11-BP04 Gestione delle informazioni

La gestione delle informazioni permette ai membri del team di trovare le informazioni necessarie per svolgere il proprio lavoro. Nella organizzazioni che promuovono la formazione dei propri dipendenti, le informazioni vengono liberamente condivise, migliorando le competenze personali. Le informazioni possono essere vagliate o cercate. Le informazioni sono accurate e aggiornate. Esistono meccanismi per creare nuove informazioni, aggiornare quelle esistenti e archiviare quelle obsolete. L'esempio più comune di una piattaforma di gestione delle informazioni è un sistema di gestione dei contenuti come un Wiki.

Risultato desiderato:

- Accesso per i membri del team a informazioni tempestive e accurate.
- Possibilità di eseguire ricerche nelle informazioni.
- Presenza di un meccanismo per aggiungere, aggiornare e archiviare le informazioni.

Anti-pattern comuni:

- Assenza di un sistema di archiviazione centrale delle informazioni. I membri del team gestiscono i propri appunti su computer locali.
- Presenza di un Wiki self-hosted, ma senza alcun meccanismo per la gestione delle informazioni, con informazioni non aggiornate di conseguenza.
- Le informazioni mancanti vengono identificate da qualcuno, ma non esiste un processo per richiederne l'aggiunta nel Wiki del team. I dipendenti le aggiungono manualmente ma omettono un passaggio importante, causando un'interruzione.

Vantaggi dell'adozione di questa best practice:

- I membri del team acquisiscono le competenze necessarie perché le informazioni vengono condivise liberamente.
- Nuovi membri del team vengono integrati più facilmente perché la documentazione è aggiornata e può essere oggetto di ricerche.
- Le informazioni sono tempestive, accurate e di utilità pratica.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

La gestione delle informazioni è un aspetto importante delle aziende che promuovono la formazione dei propri dipendenti. Per iniziare, è necessario un repository centrale in cui archiviare le informazioni, un esempio comune del quale è un Wiki self-hosted. Devi sviluppare processi per l'aggiunta, l'aggiornamento e l'archiviazione delle informazioni. Sviluppa standard per gli aspetti da documentare e permetti a ciascuno di contribuire.

Esempio del cliente

AnyCompany Retail ospita un Wiki interno in cui vengono archiviate tutte le informazioni. I membri del team sono incoraggiati ad aggiungere il proprio input nella knowledge base durante lo svolgimento

delle proprie mansioni quotidiane. Ogni trimestre un team interfunzionale valuta le pagine obsolete e determina se devono essere archiviate o aggiornate.

Passaggi dell'implementazione

1. Per iniziare, identifica il sistema di gestione dei contenuti in cui verranno archiviate le informazioni. Ottieni il consenso degli stakeholder in tutta l'organizzazione.
   a. Se non possiedi un sistema di gestione dei contenuti, valuta se eseguire un Wiki self-hosted o usare un repository con controllo delle versioni come punto di partenza.
2. Sviluppa runbook per l'aggiunta, l'aggiornamento e l'archiviazione delle informazioni. Fornisci ai team la formazione necessaria su questi processi.
3. Identifica le informazioni che devono essere archiviate nel sistema di gestione dei contenuti. Inizia dalle attività quotidiane (runbook e playbook) svolte dai membri del team. Collabora con gli stakeholder per classificare in ordine di priorità le informazioni aggiunte.
4. Collabora periodicamente con gli stakeholder per identificare le informazioni obsolete e archiviarle o aggiornarle.

Livello di impegno per il piano di implementazione: medio. Se non possiedi un sistema di gestione dei contenuti, puoi configurare un Wiki self-hosted o un repository di documenti con controllo delle versioni.

Risorse

Best practice correlate:

- OPS11-BP08 Documentazione e condivisione delle conoscenze acquisite – La gestione delle informazioni semplifica la condivisione delle conclusioni sulle lezioni apprese.

Documenti correlati:

- Atlassian - Knowledge Management

Esempi correlati:

- DokuWiki
- Gollum
- MediaWiki

- Wiki.js

OPS11-BP05 Definizione dei fattori che promuovono il miglioramento

Identifica i fattori che promuovono il miglioramento in modo da valutare e dare priorità alle opportunità sulla base di dati e cicli di feedback. Esplora le opportunità di miglioramento nei sistemi e nei processi e automatizza laddove appropriato.

Risultato desiderato:

- Tieni traccia dei dati provenienti da tutto l'ambiente.
- Esegui la correlazione di eventi e attività ai risultati aziendali.
- Puoi confrontare e contrapporre ambienti e sistemi.
- Mantieni una cronologia dettagliata delle attività relative alle implementazioni e ai risultati.
- Raccogli i dati a supporto del livello di sicurezza.

Anti-pattern comuni:

- Raccogli dati da tutto l'ambiente, ma non correli eventi e attività.
- Raccogli dati dettagliati da tutta la proprietà, aumentando l'attività e i costi di Amazon CloudWatch e AWS CloudTrail, tuttavia non utilizzi questi dati in modo significativo.
- Non tieni conto dei risultati aziendali quando definisci i fattori che promuovono il miglioramento.
- Non misuri gli effetti delle nuove funzionalità.

Vantaggi dell'adozione di questa best practice:

- Determinando i criteri di miglioramento, riduci al minimo l'impatto delle motivazioni basate sugli eventi o degli investimenti influenzati da fattori emotivi.
- Rispondi agli eventi aziendali, non solo a quelli tecnici.
- Misuri l'ambiente per identificare le aree di miglioramento.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Comprensione dei fattori che promuovono il miglioramento: è consigliabile apportare modifiche a un sistema solo quando un risultato desiderato è supportato.
  - Funzionalità desiderate: prendi in considerazione le funzionalità e le capacità desiderate quando valuti le opportunità di miglioramento.
    - Novità di AWS
  - Problemi inaccettabili: tieni in considerazione i problemi, i bug e le vulnerabilità inaccettabili quando valuti le opportunità di miglioramento. Tieni traccia delle giuste opzioni di dimensionamento e individua le opportunità di ottimizzazione.
    - Ultimi bollettini sulla sicurezza AWS
    - AWS Trusted Advisor
    - Cloud Intelligence Dashboards
  - Requisiti di conformità: quando esamini le opportunità di miglioramento, prendi in considerazione gli aggiornamenti e le modifiche necessarie per mantenere la conformità a normative e policy o per avere diritto al supporto di terze parti.
    - Conformità di AWS
    - Programmi per la conformità di AWS
    - Ultime novità sulla conformità di AWS

Risorse

Best practice correlate:

- OPS01 Priorità dell'organizzazione
- OPS02 Relazioni e proprietà
- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS08 Utilizzare l'osservabilità del carico di lavoro
- OPS09 Comprensione dello stato operativo
- OPS11-BP03 Implementazione di cicli di feedback

Documenti correlati:

- Amazon Athena

- Amazon QuickSight
- Conformità di AWS
- Ultime novità sulla conformità di AWS
- Programmi per la conformità di AWS
- AWS Glue
- Ultimi bollettini sulla sicurezza AWS
- AWS Trusted Advisor
- Esporta i dati di log in Amazon S3
- Novità di AWS
- Gli aspetti imprescindibili dell'innovazione orientata al cliente
- Digital Transformation: Hype or a Strategic Necessity?

Video correlati

- AWS re:Invent 2023 - Improve operational efficiency and resilience with AWS Support (SUP310)

OPS11-BP06 Convalida delle informazioni

Rivedi i risultati dell'analisi e le risposte con i team trasversali e i proprietari dell'azienda. Utilizza queste revisioni per definire una visione comune, identificare ulteriori impatti e stabilire le linee d'azione. Adatta le risposte, se necessario.

Risultati desiderati:

- Rivedi regolarmente gli approfondimenti con i proprietari dell'azienda. I proprietari dell'azienda forniscono un contesto aggiuntivo agli approfondimenti appena acquisiti.
- Esamini gli approfondimenti e richiedi il feedback ai colleghi tecnici, quindi condividi le tue conoscenze con i team.
- Pubblichi i dati e gli approfondimenti affinché altri team tecnici e aziendali possano esaminarli. Tieni conto dei tuoi apprendimenti nelle nuove procedure di altri reparti.
- Riassumi ed esamini i nuovi approfondimenti con i leader senior. I leader senior utilizzano i nuovi approfondimenti per definire la strategia.

Anti-pattern comuni:

- Rilasci una nuova funzionalità che modifica alcuni comportamenti dei clienti. La tua osservabilità non tiene conto di queste modifiche. Non quantifichi i vantaggi di queste modifiche.
- Effettui un nuovo aggiornamento e trascuri l'aggiornamento della rete di distribuzione di contenuti (CDN). La cache della CDN non è più compatibile con l'ultima versione. Misuri la percentuale di richieste con errori. Tutti gli utenti segnalano errori HTTP 400 durante le comunicazioni con i server di backend. Analizzi gli errori del cliente e scopri che, poiché hai misurato la dimensione sbagliata, il tuo tempo è stato improduttivo.
- L'accordo sul livello di servizio prevede un tempo di attività del 99,9% e l'obiettivo del punto di ripristino è di quattro ore. Il proprietario del servizio sostiene che il sistema non subisce tempi di inattività. Implementi una soluzione di replica costosa e complessa, che comporta uno spreco di tempo e denaro.

Vantaggi dell'adozione di questa best practice:

- Convalidando gli approfondimenti con i proprietari dell'azienda e con gli esperti in materia, è possibile stabilire una comprensione comune e gestire il miglioramento in modo più efficace.
- Individui i problemi nascosti e ne tieni conto nelle decisioni future.
- La tua attenzione si sposta dai risultati tecnici ai risultati aziendali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Convalida degli approfondimenti: interagisci con i proprietari dell'azienda e gli esperti in materia per garantire la comprensione e l'accordo comuni sul significato dei dati raccolti. Individua ulteriori problemi e impatti potenziali e stabilisci le azioni da intraprendere.

Risorse

Best practice correlate:

- OPS01-BP06 Valutazione dei compromessi gestendo vantaggi e rischi
- OPS02-BP06 Predefinizione o negoziazione delle responsabilità tra i team
- OPS11-BP03 Implementazione di cicli di feedback

Documenti correlati:

- Designing a Cloud Center of Excellence (CCOE)

Video correlati:

- Building observability to increase resiliency

OPS11-BP07 Revisione dei parametri delle operazioni

Esegui regolarmente un'analisi retrospettiva dei parametri operativi con i partecipanti di vari team da diverse aree del business. Utilizza queste revisioni per identificare opportunità di miglioramento e potenziali linee d'azione e per condividere le conoscenze acquisite. Cerca opportunità di miglioramento in tutti i tuoi ambienti, ad esempio sviluppo, test e produzione.

Risultato desiderato:

- Esamini di frequente le metriche che hanno un impatto sull'azienda.
- Rilevi ed esamini le anomalie con le tue capacità di osservabilità.
- Utilizzi i dati per supportare i risultati e gli obiettivi aziendali.

Anti-pattern comuni:

- La finestra di manutenzione interrompe un'importante promozione al dettaglio. L'azienda non è al corrente del fatto che i normali interventi di manutenzione possono essere rimandati nel caso vi siano altri eventi di particolare rilievo per l'azienda.
- Per l'uso comune di una libreria obsoleta nella tua organizzazione si è verificata una prolungata interruzione del servizio. Successivamente hai eseguito la migrazione a una libreria supportata. Gli altri team della tua organizzazione non sanno di essere a rischio.
- Non verifichi regolarmente l'aderenza agli SLA dei clienti. Le tendenze indicano un andamento negativo per quanto riguarda il rispetto degli SLA. In caso di mancato rispetto degli SLA, sono previste sanzioni economiche.

Vantaggi dell'adozione di questa best practice:

- Durante le riunioni che organizzi regolarmente per esaminare le metriche operative, gli eventi e gli incidenti, stabilisci una comprensione comune tra i team.

- Il tuo team si riunisce regolarmente per esaminare metriche e incidenti, il che ti consente di intervenire sui rischi e rispettare gli SLA dei clienti.
- Condividi le lezioni apprese, che forniscono dati per la definizione delle priorità e miglioramenti mirati per ottenere i risultati aziendali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Esegui regolarmente un'analisi retrospettiva dei parametri operativi con i partecipanti di vari team da diverse aree del business.
- Coinvolgi i soggetti interessati, compresi i team che si occupano di business, sviluppo e operazioni, per convalidare ciò che è emerso dal feedback immediato e dall'analisi retrospettiva e per condividere le conoscenze acquisite.
- Utilizza gli approfondimenti di cui dispongono per identificare opportunità di miglioramento e possibili linee d'azione.

Risorse

Best practice correlate:

- OPS08-BP05 Creare dashboard
- OPS09-BP03 Revisione delle metriche operative e assegnazione delle priorità per favorire il miglioramento
- OPS10-BP01 Utilizzo di un processo per la gestione di eventi, incidenti e problemi

Documenti correlati:

- Amazon CloudWatch
- Amazon CloudWatch metrics and dimensions reference
- Publish custom metrics
- Using Amazon CloudWatch metrics
- Dashboards and visualizations with CloudWatch

OPS11-BP08 Documentazione e condivisione delle conoscenze acquisite

Documenta e condividi le conoscenze acquisite durante le attività operative per metterle a frutto internamente e nei vari team. La condivisione di quanto appreso dai team comporta maggiori vantaggi all'interno dell'organizzazione. Condividi informazioni e risorse per impedire che si verifichino errori evitabili e semplificare le attività di sviluppo e concentrati sulla distribuzione delle funzionalità desiderate.

Utilizza AWS Identity and Access Management (IAM) per definire le autorizzazioni che consentono un accesso controllato alle risorse che desideri condividere tra i vari account.

Risultato desiderato:

- Utilizzi repository dotati di controllo versione per condividere librerie dell'applicazione, procedure di scripting, documentazione di procedure e altra documentazione di sistema.
- Condividi gli standard dell'infrastruttura come modelli AWS CloudFormation con controllo delle versioni.
- Riesamini le lezioni apprese con i team.

Anti-pattern comuni:

- Per l'uso comune di una libreria contenente degli errori nella tua organizzazione si è verificata una prolungata interruzione del servizio. Successivamente hai eseguito la migrazione a una libreria affidabile. Gli altri team della tua organizzazione non sanno di essere a rischio. Nessuno documenta e condivide l'esperienza relativa a questa libreria e nessuno è consapevole del rischio.
- Hai identificato un caso limite in un microservizio condiviso internamente che causa l'interruzione delle sessioni. Hai aggiornato le chiamate al servizio per evitare questo caso limite. Gli altri team della tua organizzazione non sanno di essere a rischio.
- Hai trovato un modo per ridurre in modo significativo i requisiti di utilizzo della CPU per uno dei tuoi microservizi. Non sai se altri team potrebbero sfruttare questa tecnica.

Vantaggi dell'adozione di questa best practice: condividi le lezioni apprese a supporto del miglioramento e per trarre il massimo vantaggio dall'esperienza.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

- Documenta e condividi le conoscenze acquisite: predisponi le procedure per documentare le conoscenze acquisite dall'esecuzione delle attività operative e dalle analisi retrospettive affinché tali informazioni possano essere utilizzate dal altri team.
- Condividi le conoscenze acquisite: predisponi le procedure per condividere con tutti i team le conoscenze acquisite e gli artefatti associati. Ad esempio condividi le procedure, le istruzioni, la governance e le best practice aggiornate tramite un wiki accessibile. Condividi script, codice e librerie tramite un repository comune.
  - Delegating access to your AWS environment
  - Share an AWS CodeCommit repository

Risorse

Best practice correlate:

- OPS02-BP06 Predefinizione o negoziazione delle responsabilità tra i team
- OPS05-BP01 Utilizzo del controllo delle versioni
- OPS05-BP06 Condivisione degli standard di progettazione
- OPS11-BP03 Implementazione di cicli di feedback
- OPS11-BP07 Revisione dei parametri delle operazioni

Documenti correlati:

- Reduce project delays with a docs-as-code solution

Video correlati:

- Delegating access to your AWS environment
- AWS Supports You | Exploring the Incident Management Tabletop Exercise

OPS11-BP09 Allocazione di tempo per i miglioramenti

Dedica tempo e risorse nei processi per rendere possibile il miglioramento incrementale continuo.

Risultato desiderato:

- Crei duplicati temporanei paralleli di ambienti per ridurre il rischio, lo sforzo e il costo della sperimentazione e dell'esecuzione di test.
- Questi ambienti duplicati possono essere utilizzati per testare le conclusioni di analisi ed esperimenti, ma anche per sviluppare e testare i miglioramenti pianificati.
- Organizzi giornate di gioco e utilizzi il servizio del Simulatore di iniezione guasti (FIS) per fornire ai team i controlli e i guardrail necessari per eseguire esperimenti in un ambiente simile a quello di produzione.

Anti-pattern comuni:

- Si è verificato un problema di prestazioni noto nel server di applicazioni. Il problema viene aggiunto al backlog, dopo l'implementazione prevista delle varie funzionalità. Se la velocità con cui vengono aggiunte le funzionalità pianificate rimane costante, il problema di prestazioni non verrà mai risolto.
- Per supportare il miglioramento continuo, autorizzi amministratori e sviluppatori a utilizzare tutto il loro tempo aggiuntivo per definire e implementare miglioramenti. I miglioramenti non vengono mai completati.
- L'accettazione operativa è stata completata e non si testano più le procedure operative.

Vantaggi dell'adozione di questa best practice: dedicando tempo e risorse nei processi, è possibile apportare miglioramenti continui e incrementali.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

- Allocazione di tempo per apportare miglioramenti: dedica tempo e risorse all'interno dei processi per rendere possibili miglioramenti graduali e continui.
- Implementa modifiche per migliorare e valutare i risultati per favorire il successo.
- Se i risultati non sono in linea con gli obiettivi e il miglioramento resta prioritario, valuta procedure d'azione alternative.
- Simula i carichi di lavoro di produzione durante le giornate di gioco e utilizza le conoscenze conseguite da queste simulazioni per migliorare.

Risorse

Best practice correlate:

- OPS05-BP08 Utilizzo di più ambienti

Video correlati:

- AWS re:Invent 2023 - Improve application resilience with AWS Fault Injection Service

Sicurezza

Il pilastro della sicurezza contempla la capacità di proteggere dati, sistemi e asset per sfruttare le tecnologie cloud in modo da migliorare la sicurezza. È possibile trovare linee guida prescrittive sull'implementazione nel Whitepaper sul principio della sicurezza.

Aree delle best practice

- Nozioni di base sulla sicurezza
- Gestione di identità e accessi
- Rilevamento
- Protezione dell'infrastruttura
- Protezione dei dati
- Risposta agli imprevisti
- Sicurezza delle applicazioni

Nozioni di base sulla sicurezza

Domanda

- SEC 1. Come gestire un carico di lavoro in sicurezza?

SEC 1. Come gestire un carico di lavoro in sicurezza?

Per gestire il carico di lavoro in modo sicuro, è necessario applicare le best practice globali a ogni area di sicurezza. Segui i requisiti e i processi definiti in termini di eccellenza operativa a livello organizzativo e di carico di lavoro e applicali a tutte le aree. Rimanere aggiornati con le raccomandazioni di AWS e del settore nonché con l'intelligence sulle minacce aiuta a sviluppare il modello di rischio e gli obiettivi di controllo. L'automazione dei processi di sicurezza, i test e la convalida permettono di dimensionare le operazioni di sicurezza.

Best practice

- SEC01-BP01 Separazione dei carichi di lavoro tramite account
- SEC01-BP02 Utente root e proprietà dell'account sicuro
- SEC01-BP03 Identificazione e convalida degli obiettivi di controllo
- SEC01-BP04 Aggiornamento continuo sulle minacce alla sicurezza e sulle raccomandazioni
- SEC01-BP05 Riduzione dell'ambito di gestione della sicurezza
- SEC01-BP06 Implementazione automatizzata dei controlli di sicurezza standard
- SEC01-BP07 Identificare le minacce e dare priorità alle mitigazioni utilizzando un modello di minaccia.
- SEC01-BP08 Valutazione e implementazione periodiche di nuovi servizi e funzionalità di sicurezza

SEC01-BP01 Separazione dei carichi di lavoro tramite account

Definisci guardrail e isolamento comuni tra ambienti (ad esempio quelli di produzione, sviluppo e test) e carichi di lavoro attraverso una strategia multi-account. La separazione a livello di account è fortemente consigliata, in quanto fornisce un solido margine di isolamento per la sicurezza, la fatturazione e l'accesso.

Risultato desiderato: una struttura di account che isola le operazioni cloud, i carichi di lavoro non correlati e gli ambienti in account separati, aumentando la sicurezza dell'infrastruttura cloud.

Anti-pattern comuni:

- Inserimento di più carichi di lavoro non correlati con diversi livelli di sensibilità dei dati nello stesso account.
- Struttura dell'unità organizzativa (UO) scarsamente definita.

Vantaggi dell'adozione di questa best practice:

- Riduzione dell'impatto in caso di accesso involontario a un carico di lavoro.
- Governance centralizzata dell'accesso a risorse, regioni e servizi AWS.
- Garanzia di sicurezza dell'infrastruttura cloud con policy e amministrazione centralizzata dei servizi di sicurezza.
- Processo automatizzato di creazione e mantenimento dell'account.
- Audit centralizzato della tua infrastruttura per la conformità e i requisiti normativi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Gli Account AWS offrono un confine di isolamento della sicurezza tra carichi di lavoro o risorse che operano a livelli di sensibilità diversi. AWS fornisce strumenti per gestire i carichi di lavoro del cloud su larga scala attraverso una strategia multi-account per sfruttare questo margine di isolamento. Per avere una guida su concetti, modelli e implementazione di una strategia multi-account su AWS, consulta Organizing Your AWS Environment Using Multiple Accounts (Organizzazione dell'ambiente AWS con l'utilizzo di account multipli).

Se si dispone di più Account AWS, gli account devono essere organizzati in una gerarchia definita da livelli di unità organizzative (UO). I controlli di sicurezza possono quindi essere organizzati e applicati alle unità organizzative e agli account membri, stabilendo controlli preventivi coerenti sugli account membri dell'organizzazione. I controlli di sicurezza sono ereditati e consentono di filtrare le autorizzazioni disponibili per gli account membro situati ai livelli inferiori di una gerarchia di unità organizzative. Un buon progetto sfrutta questa ereditarietà per ridurre il numero e la complessità delle policy di sicurezza necessarie per ottenere i controlli desiderati per ogni account membro.

AWS Organizations e AWS Control Tower sono due servizi che possono essere utilizzati per implementare e gestire questa struttura multi-account nel proprio ambiente AWS. AWS Organizations consente di organizzare gli account in una gerarchia definita da uno o più livelli di unità organizzative, ognuna delle quali contiene una serie di account membri. Le policy di controllo dei servizi consentono all'amministratore dell'organizzazione di stabilire controlli preventivi granulari sugli account dei membri, mentre AWS Config può essere utilizzato per stabilire controlli proattivi e investigativi sugli account dei membri. Molti servizi AWS si integrano con AWS Organizations per fornire controlli amministrativi delegati e per eseguire attività specifiche del servizio su tutti gli account dei membri dell'organizzazione.

Posizionato sopra AWS Organizations, AWS Control Tower fornisce un'impostazione delle best practice in un solo clic per un ambiente AWS multi-account con una zona di destinazione. La zona di destinazione è il punto di ingresso nell'ambiente multi-account stabilito da Control Tower. Control Tower offre diversi vantaggi rispetto a AWS Organizations. Tre sono i vantaggi che consentono di migliorare la governance degli account:

- Guardrail di sicurezza obbligatori integrati che vengono applicati automaticamente agli account ammessi nell'organizzazione.
- Guardrail opzionali che possono essere attivati o disattivati per un determinato insieme di unità organizzative.

- AWS Control Tower Account Factory fornisce l'implementazione automatica di account contenenti linee di base e opzioni di configurazione pre-approvate all'interno dell'organizzazione.

Passaggi dell'implementazione

1. Progettazione di una struttura organizzativa unitaria: una struttura di unità organizzative opportunamente studiata riduce l'onere di gestione necessario per creare e mantenere le policy di controllo dei servizi e gli altri controlli di sicurezza. La struttura delle unità organizzative deve essere allineata alle esigenze aziendali, alla sensibilità dei dati e alla struttura del carico di lavoro..
2. Creazione di una zona di destinazione per l'ambiente multi-account: una zona di destinazione fornisce una base di sicurezza e infrastruttura coerente da cui l'organizzazione può sviluppare, lanciare e implementare rapidamente i carichi di lavoro. Puoi utilizzare una zona di destinazione personalizzata o AWS Control Tower per orchestrare il tuo ambiente.
3. Realizzazione di guardrail: implementa guardrail di sicurezza coerenti per il tuo ambiente attraverso la zona di destinazione. AWS Control Tower offre un elenco di controlli implementabili obbligatori e facoltativi. I controlli obbligatori vengono implementati automaticamente quando si utilizza Control Tower. Esamina l'elenco dei controlli altamente consigliati e facoltativi e adotta quelli più adatti alle tue esigenze.
4. Accesso limitato a Regioni aggiunte di recente: per le nuove Regioni AWS, le risorse IAM, ad esempio utenti e ruoli, verranno propagate solo alle Regioni specificate. Questa azione può essere eseguita tramite la console quando si utilizza Control Tower oppure regolando le policy di autorizzazione IAM in AWS Organizations.
5. Presa in esame di AWS CloudFormation StackSets: StackSets consente di implementare risorse, tra cui policy, ruoli e gruppi IAM in Account AWS e Regioni differenti a partire da un modello approvato.

Risorse

Best practice correlate:

- SEC02-BP04 Fai affidamento su un provider di identità centralizzato

Documenti correlati:

- AWS Control Tower
- Linee guida sugli audit di sicurezza AWS

- IAM Best Practices(Best Practice IAM)
- Use CloudFormation StackSets to provision resources across multiple Account AWS and regions (Utilizzo di StackSet CloudFormation per il provisioning delle risorse su più account e regioni AWS)
- Organizations FAQ (Domande frequenti sulle organizzazioni)
- AWS Organizations Concetti e terminologia
- Best Practices for Service Control Policies in an AWS Organizations Multi-Account Environment (Best practice per le policy di controllo dei servizi di AWS Organizations in un ambiente multi-account)
- Guida di riferimento per la gestione degli account AWS
- Organizzazione dell'ambiente AWS con l'utilizzo di account multipli

Video correlati:

- Enable AWS adoption at scale with automation and governance (Consentire l'adozione di AWS su larga scala con l'automazione e la governance)
- Security Best Practices the Well-Architected Way
- Building and Governing Multiple Accounts using AWS Control Tower (Creazione e gestione di account multipli con AWS Control Tower)
- Enable Control Tower for Existing Organizations (Abilitare la Control Tower per le organizzazioni esistenti)

Workshop correlati:

- Control Tower Immersion Day (Giornata di approfondimento su Control Tower)

SEC01-BP02 Utente root e proprietà dell'account sicuro

L'utente root è la figura più privilegiata di un Account AWS, ha pieno accesso amministrativo a tutte le risorse dell'account e, in alcuni casi, non può essere limitato dalle policy di sicurezza. Disabilitare l'accesso programmatico all'utente root, stabilire controlli appropriati per l'utente root ed evitare l'uso di routine dell'utente root aiuta a ridurre il rischio di esposizione involontaria delle credenziali root e la conseguente compromissione dell'ambiente cloud.

Risultato desiderato: la sicurezza dell'utente root contribuisce a ridurre la possibilità che si verifichino danni accidentali o intenzionali a causa dell'uso improprio delle credenziali dell'utente root. La

creazione di controlli investigativi può anche permettere di avvisare il personale appropriato quando vengono eseguite azioni utilizzando l'utente root.

Anti-pattern comuni:

- Utilizzo dell'utente root per attività diverse da quelle che richiedono le proprie credenziali.
- Nessun test dei piani di emergenza su base regolare per verificare il funzionamento delle infrastrutture critiche, dei processi e del personale durante un'emergenza.
- Analisi limitata al tipico flusso di accesso all'account, trascurando di considerare o testare metodi alternativi di ripristino dell'account.
- Nessuna gestione di DNS, server di posta elettronica e provider telefonici come parte del perimetro di sicurezza critico, in quanto utilizzati nel flusso di recupero degli account.

Vantaggi derivanti dall'adozione di questa best practice: proteggere l'accesso all'utente root crea la certezza che le azioni del proprio account siano controllate e sottoposte a audit.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

AWS offre molti strumenti per proteggere gli account. Tuttavia, poiché alcune di queste misure non sono abilitate per impostazione predefinita, è necessario intervenire direttamente per implementarle. Queste raccomandazioni sono i passi fondamentali per mettere in sicurezza il proprio Account AWS. Durante l'implementazione di questi passaggi, è importante creare un processo di valutazione e monitoraggio continuo dei controlli di sicurezza.

Quando si crea un Account AWS per la prima volta, si inizia con una singola identità che ha accesso completo a tutti i servizi e risorse AWS presenti nell'account. Questa identità è chiamata utente root dell'Account AWS. È possibile accedere come utente root mediante l'indirizzo e-mail e la password usati per creare l'account. A causa dei livelli elevati di accesso concessi all'utente root AWS, è necessario limitare l'uso dell'utente root AWS all'esecuzione di attività che lo richiedono specificamente. Le credenziali di accesso dell'utente root devono essere tenute sotto stretta sorveglianza e l'autenticazione a più fattori (MFA) deve essere sempre abilitata per l'utente root dell'Account AWS.

Oltre al normale flusso di autenticazione per accedere all'utente root utilizzando un nome utente, una password e un dispositivo di autenticazione a più fattori (MFA), esistono flussi di recupero dell'account che consentono di accedere all'utente root dell'Account AWS grazie all'accesso

all'indirizzo e-mail e al numero di telefono associati all'account. Pertanto, è altrettanto importante proteggere l'account e-mail dell'utente root a cui vengono inviati l'e-mail di recupero e il numero di telefono associato all'account. Considerare anche le potenziali dipendenze circolari, quando l'indirizzo e-mail associato all'utente root è ospitato su server di posta elettronica o su risorse del servizio dei nomi di dominio (DNS) dello stesso Account AWS.

Quando si utilizza AWS Organizations, esistono più Account AWS, ognuno dei quali ha un utente root. Un account è designato come account di gestione e sotto l'account di gestione possono essere aggiunti diversi livelli di account membri. La priorità è proteggere l'utente root dell'account di gestione, quindi occuparsi degli utenti root degli account membri. La strategia per la protezione dell'utente root dell'account di gestione può essere diversa da quella degli utenti root degli account membri ed è possibile effettuare controlli di sicurezza preventivi sugli utenti root degli account membri.

Passaggi dell'implementazione

Per stabilire i controlli per l'utente root si consigliano le seguenti fasi di implementazione. Eventuali raccomandazioni sono collegate a CIS AWS Foundations benchmark versione 1.4.0. Oltre a questi passaggi, consulta le AWS best practice guidelines (Linee guida sulle best practice AWS) per la protezione delle risorse e degli Account AWS.

Controlli preventivi

1. Imposta informazioni di contatto accurate per l'account.
    a. Queste informazioni vengono utilizzate per il flusso di recupero della password persa, per il flusso di recupero dell'account del dispositivo MFA perso e per le comunicazioni critiche relative alla sicurezza con il team.
    b. Utilizza un indirizzo e-mail ospitato dal dominio aziendale, preferibilmente una lista di distribuzione, come indirizzo e-mail dell'utente root. L'utilizzo di una lista di distribuzione piuttosto che dell'account di e-mail di un singolo individuo offre una maggiore ridondanza e continuità di accesso all'account root per lunghi periodi di tempo.
    c. Il numero di telefono indicato nelle informazioni di contatto deve essere dedicato e sicuro per questo scopo. Il numero di telefono non deve essere indicato o condiviso con nessuno.
2. Non creare chiavi di accesso per l'utente root. Se sono presenti chiavi di accesso, rimuovile (CIS 1.4).
    a. Elimina le credenziali programmatiche a lunga durata (chiavi di accesso e segrete) per l'utente root.

   b. Se esistono già chiavi di accesso per l'utente root, è necessario passare i processi che utilizzano tali chiavi all'uso di chiavi di accesso temporanee di un ruolo AWS Identity and Access Management (IAM), quindi eliminare le chiavi di accesso per l'utente root.

3. Stabilisci se è necessario memorizzare le credenziali per l'utente root.

   a. Se utilizzi AWS Organizations per creare nuovi account membro, la password iniziale dell'utente root sui nuovi account membro è impostata su un valore casuale che non è visibile a te. Considera l'utilizzo del flusso di ripristino della password dal tuo account di gestione di AWS Organization per ottenere l'accesso all'account membro, se necessario.

   b. Per gli Account AWS standalone o per l'account di gestione di AWS Organization, considera la creazione e l'archiviazione sicura delle credenziali per l'utente root. Abilita la MFA per l'utente root.

4. Abilita i controlli preventivi per gli utenti root degli account membri in ambienti multi-account AWS.

   a. Considera di attivare il guardrail preventivo Disallow Creation of Root Access Keys for the Root User (Disabilitare la creazione di chiavi di accesso root per l'utente root) per gli account dei membri.

   b. Considera di attivare il guardrail preventivo Disallow Actions as a Root User (Disabilitare le azioni come utente root) per gli account dei membri.

5. Se sono necessarie le credenziali per l'utente root:

   a. Utilizza una password complessa.

   b. Abilita l'autenticazione a più fattori (MFA) per l'utente root, in particolare per gli account dei manager (paganti) AWS Organizations (CIS 1.5).

   c. Considera i dispositivi MFA hardware per la resilienza e la sicurezza, in quanto i dispositivi monouso possono ridurre le possibilità che i dispositivi contenenti i codici MFA vengano riutilizzati per altri scopi. Verifica che i dispositivi hardware MFA alimentati da una batteria siano sostituiti regolarmente. (CIS 1.6)

      - Per configurare l'MFA per l'utente root, segui le istruzioni per abilitare un dispositivo MFA o virtuale o hardware.

   d. Considera la possibilità di iscrivere più dispositivi MFA per il backup. Sono consentiti fino a 8 dispositivi MFA per account.

      - Tieni presente che l'iscrizione di più di un dispositivo MFA per l'utente root disabilita automaticamente il flusso per il recupero dell'account in caso di perdita del dispositivo MFA..

e. Conserva la password in modo sicuro e considera le dipendenze circolari se la password viene conservata elettronicamente. Non memorizzare la password in modo tale da richiedere l'accesso allo stesso Account AWS per ottenerla.

6. Facoltativo: valuta la possibilità di stabilire un programma di rotazione periodica delle password per l'utente root.
   - Le best practice per la gestione delle credenziali dipendono dai requisiti normativi e di policy. Gli utenti root protetti da MFA non dipendono dalla password come unico fattore di autenticazione.
   - La modifica della password dell'utente root su base periodica riduce il rischio che una password esposta inavvertitamente possa essere utilizzata in modo improprio.

Controlli di rilevamento

- Crea allarmi per rilevare l'uso delle credenziali root (CIS 1.7). L'abilitazione di Amazon GuardDuty monitorerà e segnalerà l'uso delle credenziali API dell'utente root attraverso il rilevamento RootCredentialUsage.
- Valuta e implementa i controlli investigativi inclusi in AWS Well-Architected Security Pillar conformance pack for AWS Config (Pacchetto di conformità del pilastro di sicurezza well-architected di AWS per AWS Config) oppure, se si utilizza AWS Control Tower, i controlli fortemente consigliati disponibili in Control Tower.

Guida operativa

- Stabilisci chi nell'organizzazione deve avere accesso alle credenziali dell'utente root.
  - Utilizza una regola a due persone, in modo che nessun individuo abbia accesso a tutte le credenziali necessarie e all'MFA per ottenere l'accesso come utente root.
  - Verifica che l'organizzazione, e non un singolo individuo, mantenga il controllo sul numero di telefono e sull'alias e-mail associati all'account (utilizzati per il ripristino della password e il flusso di ripristino MFA).
- Utilizza l'utente root solo in via eccezionale (CIS 1.7).
  - L'utente root dell’account AWS non deve essere utilizzato per le attività giornaliere e nemmeno per quelle amministrative. Effettua il login come utente root solo per eseguire attività AWS che lo richiedono. Tutte le altre azioni devono essere eseguite da altri utenti che assumono i ruoli appropriati.

- Verifica periodicamente che l'accesso all'utente root sia funzionante, in modo da testare le procedure prima di una situazione di emergenza che richieda l'uso delle credenziali dell'utente root.
- Verifica periodicamente che l'indirizzo e-mail associato all'account e quelli elencati in Alternate Contacts (Contatti alternativi) funzionino. Monitora queste caselle di posta elettronica per le notifiche di sicurezza che potresti ricevere da <abuse@amazon.com>. Assicurati inoltre che i numeri di telefono associati all'account siano attivi.
- Prepara procedure di risposta agli incidenti per rispondere all'uso improprio dell'account root. Consulta la AWS Security Incident Response Guide (Guida alla risposta agli incidenti di sicurezza AWS) e alle best practice riportate nella sezione Incident Response (Risposta agli incidenti) del whitepaper Security Pillar (Pilastro di sicurezza) per ulteriori informazioni sulla creazione di una strategia di risposta agli incidenti del tuo Account AWS.

Risorse

Best practice correlate:

- SEC01-BP01 Separazione dei carichi di lavoro tramite account
- SEC02-BP01 Utilizzo meccanismi di accesso efficaci
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- SEC03-BP03 Determinazione di un processo per l'accesso di emergenza
- SEC10-BP05 Preassegnazione dell'accesso

Documenti correlati:

- AWS Control Tower
- Linee guida sugli audit di sicurezza AWS
- IAM Best Practices(Best Practice IAM)
- Amazon GuardDuty – root credential usage alert (Amazon GuardDuty – Avviso sull'utilizzo delle credenziali root)
- Step-by-step guidance on monitoring for root credential use through CloudTrail (Guida passo-passo sul monitoraggio dell'uso delle credenziali root tramite CloudTrail)
- Token MFA approvati per l'uso con AWS
- Implementazione di funzionalità di break glass access (accesso di emergenza) su AWS

- Top 10 security items to improve in your Account AWS (I 10 principali elementi di sicurezza da migliorare nel proprio account AWS)
- Che cosa devo fare se noto un'attività non autorizzata nel mio Account AWS?

Video correlati:

- Enable AWS adoption at scale with automation and governance (Consentire l'adozione di AWS su larga scala con l'automazione e la governance)
- Security Best Practices the Well-Architected Way
- Limiting use of AWS root credentials (Restrizioni nell'uso delle credenziali AWS) da AWS re:inforce 2022 – Security best practices with AWS IAM (Best practice di sicurezza con AWS IAM)

Esempi e laboratori correlati:

- Laboratorio: Account AWS e utente root

SEC01-BP03 Identificazione e convalida degli obiettivi di controllo

In base ai requisiti di conformità e ai rischi identificati dal modello di rischio, deriva e convalida gli obiettivi di controllo e i controlli da applicare al carico di lavoro. La convalida continua degli obiettivi di controllo e dei controlli aiuta a misurare l'efficacia della mitigazione dei rischi.

Risultato desiderato: gli obiettivi di controllo della sicurezza della tua azienda sono ben definiti e allineati ai tuoi requisiti di conformità. I controlli vengono implementati e applicati attraverso l'automazione e le policy e vengono costantemente valutati per verificarne l'efficacia nel raggiungimento degli obiettivi. Le prove dell'efficacia, sia in un determinato momento che in un determinato periodo di tempo, sono prontamente comunicate ai revisori.

Anti-pattern comuni:

- I requisiti normativi, le aspettative del mercato e gli standard di settore per una sicurezza certa non sono ben compresi dalla tua azienda.
- I framework di sicurezza informatica e gli obiettivi di controllo non sono allineati ai requisiti dell'azienda.
- L'implementazione dei controlli non è perfettamente allineata agli obiettivi di controllo in modo misurabile.
- L'automazione non viene utilizzata per creare report sull'efficacia dei tuoi controlli.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I framework di sicurezza informatica comunemente utilizzati sono molti e possono costituire la base per gli obiettivi di controllo della sicurezza. Per determinare quale sia il framework più adatto alle tue esigenze, considera i requisiti normativi, le aspettative del mercato e gli standard di settore dell'azienda. A titolo esemplificativo, è possibile citare AICPA SOC 2, HITRUST, PCI-DSS, ISO 27001 e NIST SP 800-53.

Per gli obiettivi di controllo identificati, occorre comprendere in che modo i servizi AWS utilizzati permettono di conseguirli. Utilizza AWS Artifact per ricercare la documentazione e i report allineati ai framework di riferimento che descrivono l'ambito di responsabilità coperto da AWS e le linee guida per l'ambito che rimane di tua competenza. Per ulteriori indicazioni specifiche sui servizi che si allineano alle varie dichiarazioni di controllo dei framework, consulta AWS Customer Compliance Guides.

Nel definire i controlli che raggiungono i tuoi obiettivi, codifica l'applicazione utilizzando i controlli preventivi e automatizza le mitigazioni mediante i controlli di rilevamento. Aiuta a prevenire le configurazioni e le azioni non conformi su AWS Organizations utilizzando le policy di controllo dei servizi. Implementa le regole in AWS Config per monitorare e segnalare le risorse non conformi, per poi passare a un modello di applicazione delle regole nel momento in cui il comportamento di tali risorse sarà sicuro. Per distribuire set di regole predefinite e gestite che si allineano ai tuoi framework di sicurezza informatica, valuta l'uso degli standard AWS Security Hub come prima opzione. Lo standard AWS Foundational Service Best Practices (FSBP) e il CIS AWS Foundations Benchmark sono validi punti di partenza con controlli che si allineano a molti obiettivi condivisi da più framework standard. Laddove Security Hub non disponga intrinsecamente dei rilevamenti di controllo desiderati, può essere integrato utilizzando i pacchetti di conformità AWS Config.

Utilizza i Pacchetti APN Partner consigliati dal team AWS Global Security and Compliance Acceleration (GSCA) per ricevere l'assistenza di consulenti di sicurezza, agenzie di consulenza, sistemi di raccolta delle prove e di reporting, revisori dei conti e altri servizi complementari, se necessario.

Passaggi dell'implementazione

1. Valuta i framework di sicurezza informatica comuni e allinea i tuoi obiettivi di controllo a quelli scelti.

2. Ottieni la documentazione pertinente sulle linee guida e le responsabilità per il tuo framework utilizzando AWS Artifact. Comprendi quali parti della conformità rientrano nel modello di responsabilità condivisa AWS e quali sono di tua competenza.
3. Utilizza le policy di controllo dei servizi, le policy sulle risorse, le policy di attendibilità dei ruoli e altri guardrail per prevenire configurazioni e azioni delle risorse non conformi.
4. Valuta l'implementazione di standard Security Hub e pacchetti di conformità AWS Config in linea con i tuoi obiettivi di controllo.

Risorse

Best practice correlate:

- SEC03-BP01 Definizione dei requisiti di accesso
- SEC04-BP01 Configurazione dei registri di servizi e applicazioni
- SEC07-BP01 Comprendere lo schema di classificazione dei dati
- OPS01-BP03 Valutazione dei requisiti di governance
- OPS01-BP04 Valutazione dei requisiti di conformità
- PERF01-BP05 Uso delle policy e delle architetture di riferimento
- COST02-BP01 Sviluppo di policy basate sui requisiti dell'organizzazione

Documenti correlati:

- AWS Customer Compliance Guides

Strumenti correlati:

- AWS Artifact

SEC01-BP04 Aggiornamento continuo sulle minacce alla sicurezza e sulle raccomandazioni

Rimani aggiornato sulle minacce più recenti e sulle misure di mitigazione monitorando le pubblicazioni di intelligence sulle minacce del settore e i feed di dati per gli aggiornamenti. Valuta le offerte di servizi gestiti che si aggiornano automaticamente in base ai dati sulle minacce più recenti.

Risultato desiderato: rimani informato man mano che le pubblicazioni del settore vengono aggiornate con le minacce e le raccomandazioni più recenti. L'automazione viene utilizzata per rilevare

potenziali vulnerabilità ed esposizioni man mano che si identificano nuove minacce. Intraprendi azioni di mitigazione contro queste minacce. Adotta servizi AWS che si aggiornano automaticamente con le informazioni sulle minacce più recenti.

Anti-pattern comuni:

- Non disporre di un meccanismo affidabile e ripetibile per rimanere informati sulle ultime informazioni sulle minacce.
- Mantenere un inventario manuale del portafoglio tecnologico, dei carichi di lavoro e delle dipendenze che richiedono un esame umano per individuare potenziali vulnerabilità ed esposizioni.
- Non disporre di meccanismi per aggiornare i carichi di lavoro e le dipendenze alle ultime versioni disponibili, che forniscono mitigazioni note delle minacce.

Vantaggi dell'adozione di questa best practice: l'utilizzo di fonti di informazioni sulle minacce per rimanere aggiornati riduce il rischio di perdere importanti cambiamenti nel panorama delle minacce che possono avere un impatto sulla propria azienda. L'automazione in atto per scansionare, rilevare e correggere eventuali vulnerabilità o esposizioni nei carichi di lavoro e nelle relative dipendenze può aiutarti a mitigare i rischi in modo rapido e prevedibile, rispetto alle alternative manuali. Questo aiuta a controllare i tempi e i costi relativi alla mitigazione delle vulnerabilità.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Consulta le pubblicazioni di intelligence sulle minacce per costanti aggiornamenti sul panorama delle minacce. Consulta la knowledge base MITRE ATT&CK per la documentazione su tattiche, tecniche e procedure contraddittorie note (TTP). Consulta l'elenco delle vulnerabilità ed esposizioni comuni (CVE) di MITRE per avere aggiornamenti sulle vulnerabilità note nei prodotti su cui fai affidamento. Comprendi i rischi critici per le applicazioni Web con il popolare progetto OWASP Top 10 dell'Open Worldwide Application Security Project (OWASP).

Ricevi aggiornamenti sugli eventi di sicurezza AWS e sulle misure correttive consigliate con i bollettini sulla sicurezza AWS per i CVE.

Per ridurre gli sforzi complessivi e il sovraccarico per rimanere aggiornati, valuta la possibilità di utilizzare i servizi AWS che incorporano automaticamente nuove informazioni sulle minacce nel tempo. Ad esempio, Amazon GuardDuty rimane aggiornato con le informazioni sulle minacce del settore per rilevare comportamenti anomali e firme delle minacce all'interno dei tuoi account.

Amazon Inspector mantiene automaticamente aggiornato un database dei CVE che utilizza per le sue funzionalità di scansione continua. AWS WAF e AWS Shield Advanced forniscono gruppi di regole gestiti che vengono aggiornati automaticamente man mano che emergono nuove minacce.

Rivedi Well-Architected operational excellence pillar per la gestione e l'applicazione automatizzate delle patch della flotta.

Passaggi dell'implementazione

- Abbonati agli aggiornamenti per le pubblicazioni di intelligence sulle minacce pertinenti alla tua azienda e al tuo settore. Iscriviti ai bollettini sulla sicurezza AWS.
- Prendi in considerazione l'adozione di servizi che incorporino automaticamente nuove informazioni sulle minacce, come Amazon GuardDuty e Amazon Inspector.
- Implementa una strategia di gestione e patching della flotta in linea con le best practice del Well-Architected Operational Excellence Pillar.

Risorse

Best practice correlate:

- SEC01-BP07 Identificare le minacce e dare priorità alle mitigazioni utilizzando un modello di minaccia.
- OPS01-BP05 Valutazione del panorama delle minacce
- OPS11-BP01 Definizione di un processo per il miglioramento continuo

SEC01-BP05 Riduzione dell'ambito di gestione della sicurezza

Stabilisci se è possibile ridurre l'ambito della sicurezza utilizzando servizi AWS che trasferiscono la gestione di alcuni controlli ad AWS (servizi gestiti). Questi servizi possono contribuire a ridurre le attività di manutenzione della sicurezza, come il provisioning dell'infrastruttura, l'impostazione del software, il patching o i backup.

Risultato desiderato: quando scegli i servizi AWS per il tuo carico di lavoro, tieni conto dell'ambito della gestione della sicurezza. Il costo delle spese generali di gestione e delle attività di manutenzione (il costo totale di proprietà o TCO) viene confrontato con il costo dei servizi selezionati, oltre ad altre considerazioni Well-Architected. La documentazione di controllo e conformità AWS viene incorporata nelle procedure di valutazione e verifica dei controlli.

Anti-pattern comuni:

- Implementazione dei carichi di lavoro senza comprendere a fondo il modello di responsabilità condivisa per i servizi selezionati.
- Hosting di database e altre tecnologie su macchine virtuali senza aver valutato un servizio gestito equivalente.
- Mancata inclusione delle attività di gestione della sicurezza nel costo totale di proprietà delle tecnologie di hosting su macchine virtuali rispetto alle opzioni di servizio gestito.

Vantaggi della definizione di questa best practice: l'utilizzo di servizi gestiti può ridurre l'onere complessivo della gestione dei controlli di sicurezza operativi, riducendo così i rischi per la sicurezza e il costo totale di proprietà. Il tempo che altrimenti sarebbe dedicato a determinate attività di sicurezza può essere reinvestito in attività che forniscono maggior valore alla tua azienda. I servizi gestiti possono anche ridurre l'ambito dei requisiti di conformità spostando alcuni requisiti di controllo su AWS.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Le modalità di integrazione dei componenti del carico di lavoro su AWS sono molteplici. L'installazione e l'esecuzione di tecnologie sulle istanze Amazon EC2 impongono spesso all'utente di assumersi la maggior parte delle responsabilità in materia di sicurezza. Per ridurre l'onere della gestione di alcuni controlli, individua i servizi gestiti AWS in grado di ridurre l'ambito della tua parte del modello di responsabilità condivisa e cerca di capire come utilizzarli nell'architettura esistente. Tra gli esempi è possibile utilizzare Amazon Relational Database Service (Amazon RDS) per l'implementazione dei database, Amazon Elastic Kubernetes Service (Amazon EKS) o Amazon Elastic Container Service (Amazon ECS) per l'orchestrazione dei container o l'utilizzo di opzioni serverless. Quando sviluppi nuove applicazioni, pensa a quali servizi possono contribuire a ridurre i tempi e i costi di implementazione e gestione dei controlli di sicurezza.

Anche i requisiti di conformità possono essere un fattore di scelta dei servizi. I servizi gestiti possono trasferire la conformità di alcuni requisiti ad AWS. Discuti con il tuo team di conformità riguardo al loro livello di familiarità nel sottoporre a audit gli aspetti dei servizi che gestisci e nell'accettare le dichiarazioni di controllo nei relativi report di audit di AWS. Puoi fornire gli artefatti di audit trovati in AWS Artifact ai tuoi revisori o autorità di regolamentazione come prova dei controlli di sicurezza AWS. Puoi anche utilizzare le linee guida sulla responsabilità fornite da alcuni degli artefatti di audit AWS per progettare la tua architettura, insieme alle AWS Customer Compliance Guides. Queste

indicazioni aiutano a determinare i controlli di sicurezza aggiuntivi da mettere in atto per supportare i casi d'uso specifici del sistema.

Quando utilizzi servizi gestiti, è bene conoscere il processo di aggiornamento delle loro risorse a versioni più recenti (ad esempio, l'aggiornamento della versione di un database gestito da Amazon RDS o del runtime del linguaggio di programmazione per una funzione AWS Lambda). Anche se il servizio gestito può eseguire questa operazione per tuo conto, la configurazione della tempistica dell'aggiornamento e la conoscenza dell'impatto sulle tue operazioni restano di tua responsabilità. Strumenti come AWS Health possono aiutarti a tracciare e gestire questi aggiornamenti in tutti i tuoi ambienti.

Passaggi dell'implementazione

1. Valuta i componenti del tuo carico di lavoro che possono essere sostituiti con un servizio gestito.
    a. Se stai migrando un carico di lavoro ad AWS, considera la riduzione della gestione (tempo e spese) e la conseguente diminuzione del rischio quando valuti l'opportunità di rehosting, rifattorizzazione, ridefinizione della piattaforma, ricostruzione o sostituzione del carico di lavoro. A volte un investimento aggiuntivo all'inizio di una migrazione può comportare risparmi significativi nel lungo periodo.
2. Prendi in considerazione l'implementazione di servizi gestiti, ad esempio Amazon RDS, invece di installare e gestire le tue implementazioni tecnologiche.
3. Utilizza le linee guida sulla responsabilità in AWS Artifact per definire i controlli di sicurezza da adottare per il tuo carico di lavoro.
4. Tieni un inventario delle risorse in uso e rimani aggiornato con nuovi servizi e approcci per identificare nuove opportunità per ridurre l'ambito.

Risorse

Best practice correlate:

- PERF02-BP01 Selezione delle migliori opzioni di elaborazione per il carico di lavoro
- PERF03-BP01 Uso di un archivio dati dedicato che supporta al meglio i requisiti di accesso e archiviazione dei dati
- SUS05-BP03 Utilizzo dei servizi gestiti

Documenti correlati:

- Planned lifecycle events for AWS Health

Strumenti correlati:

- AWS Health
- AWS Artifact
- AWS Customer Compliance Guides

Video correlati:

- How do I migrate to an Amazon RDS or Aurora MySQL DB instance using AWS DMS?
- AWS re:Invent 2023 - Manage resource lifecycle events at scale with AWS Health

SEC01-BP06 Implementazione automatizzata dei controlli di sicurezza standard

Applica pratiche DevOps moderne mentre sviluppi e distribuisci controlli di sicurezza standard in tutti i tuoi ambienti AWS. Definisci controlli e configurazioni di sicurezza standard utilizzando i modelli Infrastructure as Code (IaC), acquisisci le modifiche in un sistema di controllo della versione, testa le modifiche come parte di una pipeline CI/CD e automatizza l'implementazione delle modifiche nei tuoi ambienti AWS.

Risultato desiderato: i modelli IaC acquisiscono controlli di sicurezza standardizzati e li affidano a un sistema di controllo della versione. Le pipeline CI/CD si trovano in luoghi che rilevano le modifiche e automatizzano i test e l'implementazione degli ambienti AWS. Sono presenti dei guardrail per rilevare e avvisare in caso di configurazioni errate nei modelli prima di procedere all'implementazione. I carichi di lavoro vengono distribuiti in ambienti in cui sono presenti controlli standard. I team hanno accesso all'implementazione di configurazioni di servizio approvate tramite un meccanismo self-service. Sono disponibili strategie di backup e ripristino sicure per le configurazioni di controllo, gli script e i dati correlati.

Anti-pattern comuni:

- Apportare modifiche ai controlli di sicurezza standard manualmente, tramite una console Web o un'interfaccia a riga di comando.
- Affidarsi ai singoli team del carico di lavoro per implementare manualmente i controlli definiti da un team centrale.

- Affidarsi a un team di sicurezza centrale per implementare i controlli a livello di carico di lavoro su richiesta di un team del carico di lavoro.
- Consentire agli stessi individui o team di sviluppare, testare e implementare script di automazione per il controllo della sicurezza senza un'adeguata separazione dei compiti o dei controlli e degli equilibri.

Vantaggi della definizione di questa best practice: l'utilizzo di modelli per definire i controlli di sicurezza standard consente di tracciare e confrontare le modifiche nel tempo utilizzando un sistema di controllo delle versioni. L'uso dell'automazione per testare e implementare le modifiche crea standardizzazione e prevedibilità, aumentando le possibilità di una corretta implementazione e riducendo le attività manuali ripetitive. Fornire un meccanismo self-service per consentire ai team addetti al carico di lavoro di implementare servizi e configurazioni approvati riduce il rischio di configurazioni errate e usi impropri. Questo li aiuta anche a incorporare i controlli nelle prime fasi del processo di sviluppo.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Quando si seguono le pratiche descritte in SEC01-BP01 Carichi di lavoro separati utilizzando account, si ottengono più Account AWS per ambienti diversi che è possibile gestire utilizzando AWS Organizations. Sebbene ciascuno di questi ambienti e carichi di lavoro possa richiedere controlli di sicurezza distinti, puoi standardizzare alcuni controlli di sicurezza in tutta l'organizzazione. Gli esempi includono l'integrazione di provider di identità centralizzati, la definizione di reti e firewall e la configurazione di posizioni standard per l'archiviazione e l'analisi dei log. Allo stesso modo in cui è possibile utilizzare Infrastructure as code (IaC) per applicare lo stesso rigore dello sviluppo del codice applicativo al provisioning dell'infrastruttura, è possibile utilizzare IaC anche per definire e implementare i controlli di sicurezza standard.

Ove possibile, definisci i tuoi controlli di sicurezza in modo dichiarativo, ad esempio in AWS CloudFormation, e memorizzali in un sistema di controllo del codice origine. Usa le pratiche DevOps per automatizzare l'implementazione dei controlli per versioni più prevedibili, i test automatici utilizzando strumenti come AWS CloudFormation Guard e il rilevamento della deriva tra i controlli distribuiti e la configurazione desiderata. È possibile utilizzare servizi come AWS CodePipeline, AWS CodeBuild e AWS CodeDeploy per creare una pipeline CI/CD. Prendi in considerazione le indicazioni contenute in Organizing your AWS Environment Using Multiple Accounts per configurare questi servizi nei propri account separati dalle altre pipeline di implementazione.

Puoi inoltre definire modelli per standardizzare la definizione e l'implementazione di Account AWS, servizi e configurazioni. Questa tecnica consente a un team di sicurezza centrale di gestire queste definizioni e di fornirle ai team che si occupano dei carichi di lavoro attraverso un approccio self-service. Un modo per raggiungere questo obiettivo è utilizzare Service Catalog, dove è possibile pubblicare modelli come prodotti che i team del carico di lavoro possono incorporare nelle proprie implementazioni di pipeline. Se si utilizza AWS Control Tower, alcuni modelli e controlli sono disponibili come punto di partenza. Control Tower offre anche la funzionalità Account Factory, che consente ai team addetti al carico di lavoro di creare nuovi Account AWS utilizzando gli standard definiti dall'utente. Questa funzionalità aiuta a rimuovere le dipendenze da un team centrale per l'approvazione e la creazione di nuovi account quando vengono identificati come necessari dai team del carico di lavoro. Potresti aver bisogno di questi account per isolare i diversi componenti del carico di lavoro in base a motivi quali la funzione che svolgono, la sensibilità dei dati elaborati o il loro comportamento.

Passaggi dell'implementazione

1. Determina come archivierai e manterrai i tuoi modelli in un sistema di controllo della versione.
2. Crea pipeline CI/CD per testare e distribuire i tuoi modelli. Definisci i test per verificare che non ci siano configurazioni errate e che i modelli siano conformi agli standard aziendali.
3. Crea un catalogo di modelli standardizzati affinché i team addetti al carico di lavoro possano implementare Account AWS e fornire servizi in base alle tue esigenze.
4. Implementa strategie di backup e ripristino sicure per le configurazioni di controllo, gli script e i dati correlati.

Risorse

Best practice correlate:

- OPS05-BP01 Utilizzo del controllo delle versioni
- OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione
- REL08-BP05 Implementazione delle modifiche tramite automazione
- SUS06-BP01 Adozione di metodi che consentano di introdurre rapidamente migliorie in tema di sostenibilità

Documenti correlati:

- Organizing Your AWS Environment Using Multiple Accounts

Esempi correlati:

- Automate account creation, and resource provisioning using Service Catalog, AWS Organizations, and AWS Lambda
- Strengthen the DevOps pipeline and protect data with AWS Secrets Manager, AWS KMS, and AWS Certificate Manager

Strumenti correlati:

- AWS CloudFormation Guard
- Landing Zone Accelerator on AWS

SEC01-BP07 Identificare le minacce e dare priorità alle mitigazioni utilizzando un modello di minaccia.

Effettua la modellazione delle minacce per identificare e mantenere un registro aggiornato delle minacce potenziali e delle relative mitigazioni per il carico di lavoro. Definisci le priorità delle minacce e adatta le mitigazioni dei controlli di sicurezza per prevenire, intercettare e rispondere. Rivedi e mantieni questo aspetto nel contesto del tuo carico di lavoro e dell'evoluzione del panorama della sicurezza.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Che cos'è la modellazione delle minacce?

"La modellazione delle minacce ha lo scopo di identificare, comunicare e comprendere le minacce e le mitigazioni nel contesto della protezione di qualcosa di valore." – The Open Web Application Security Project (OWASP) Application Threat Modeling

Perché realizzare un modello di minaccia?

I sistemi sono complessi e nel tempo diventano sempre più complessi e capaci di fornire un maggiore valore aziendale e una maggiore soddisfazione e coinvolgimento dei clienti. Ciò significa che le decisioni di progettazione IT devono tenere conto di un numero sempre maggiore di casi d'uso. Questa complessità e il numero di combinazioni di casi d'uso rendono in genere gli approcci non strutturati inefficaci per individuare e mitigare le minacce. È invece necessario un approccio sistematico per enumerare le potenziali minacce al sistema e per elaborare le mitigazioni e stabilirne

le priorità per assicurarsi che le risorse limitate dell'organizzazione abbiano il massimo impatto nel migliorare lo stato di sicurezza complessiva del sistema.

La modellazione delle minacce è progettata per fornire questo approccio sistematico, con l'obiettivo di trovare e affrontare i problemi nelle prime fasi del processo di progettazione, quando le mitigazioni hanno un costo e un impegno relativi bassi rispetto alle fasi successive del ciclo di vita. Questo approccio è in linea con il principio di sicurezza shift-left del settore. In definitiva, la modellazione delle minacce si integra con il processo di gestione del rischio di un'organizzazione e aiuta a prendere decisioni sui controlli da implementare utilizzando un approccio orientato alle minacce.

Quando è necessario eseguire la modellazione delle minacce?

La modellazione delle minacce deve essere avviata il più presto possibile nel ciclo di vita del carico di lavoro, in modo da avere una maggiore flessibilità di intervento sulle minacce identificate. Come per i bug del software, prima si identificano le minacce, più è conveniente affrontarle. Un modello di minacce è un documento vivo e deve continuare a evolvere in base ai cambiamenti dei carichi di lavoro. I modelli di minaccia vanno rivisti nel tempo, anche in caso di modifiche importanti, di cambiamenti nel panorama delle minacce o di adozione di nuove funzionalità o servizi.

Passaggi dell'implementazione

Come possiamo eseguire la modellazione delle minacce?

Esistono diversi modi per eseguire la modellazione delle minacce. Come per i linguaggi di programmazione, anche in questo caso ci sono vantaggi e svantaggi e bisogna scegliere il metodo più adatto alle proprie esigenze. Un approccio possibile è iniziare con Shostack’s 4 Question Frame for Threat Modeling, che pone domande aperte per strutturare l'esercizio di modellazione delle minacce:

1. A cosa si sta lavorando?

   Questa domanda ha lo scopo di aiutare a comprendere e concordare il sistema che si sta costruendo e i dettagli di tale sistema che sono rilevanti per la sicurezza. La creazione di un modello o di un diagramma è il modo più diffuso per rispondere a questa domanda, in quanto aiuta a visualizzare ciò che si sta costruendo, ad esempio utilizzando un diagramma di flusso dei dati. Scrivere le ipotesi e i dettagli importanti del sistema aiuta anche a definire l'ambito di applicazione. In questo modo, tutti coloro che contribuiscono alla modellazione delle minacce possono concentrarsi sullo stesso aspetto, evitando deviazioni dispendiose in termini di tempo su argomenti fuori portata (comprese le versioni non aggiornate del sistema). Ad esempio, se si sta costruendo un'applicazione web, probabilmente non vale la pena procedere alla modellazione

per la sequenza di avvio attendibile del sistema operativo per i browser client, poiché non si ha la possibilità di influire su questo aspetto con il proprio progetto.

2. Cosa può andare storto?

   In questa fase si identificano le minacce al sistema. Le minacce sono azioni o eventi accidentali o intenzionali che hanno impatti indesiderati e potrebbero compromettere la sicurezza del sistema. Senza una visione chiara di ciò che potrebbe andare storto, non è possibile fare nulla per evitarlo.

   Non esiste un elenco canonico di ciò che può andare storto. La creazione di questo elenco richiede un brainstorming e la collaborazione di tutti i componenti del team e dei soggetti coinvolti nell'esercizio di modellazione delle minacce. Per facilitare il brainstorming si può utilizzare un modello per l'identificazione delle minacce, ad esempio STRIDE, che suggerisce diverse categorie da valutare: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of privilege (spoofing, manomissione, ripudio, divulgazione di informazioni, negazione del servizio ed elevazione dei privilegi). Inoltre, per facilitare il brainstorming, si possono consultare gli elenchi e le ricerche esistenti per trarne ispirazione, come ad esempio OWASP Top 10, HiTrust Threat Catalog e il catalogo delle minacce della propria organizzazione.

3. Che cosa faremo a questo proposito?

   Come nel caso della domanda precedente, non esiste un elenco canonico di tutte le possibili mitigazioni. Gli input di questa fase sono le minacce, gli attori e le aree di miglioramento identificate nella fase precedente.

   La sicurezza e la conformità sono una responsabilità condivisa da AWS e dal cliente. È importante capire che quando si chiede "Che cosa faremo?", si chiede anche "Chi è responsabile? Chi ha la responsabilità di fare qualcosa?" Comprendere l'equilibrio delle responsabilità tra utente e AWS consente di limitare l'esercizio di modellazione delle minacce alle mitigazioni sotto il proprio controllo, che di solito sono una combinazione di opzioni di configurazione del servizio AWS e di mitigazioni specifiche del proprio sistema.

   Per la parte AWS relativa alla responsabilità condivisa, si scoprirà che i servizi AWS rientrano nell'ambito di molti programmi di conformità. Questi programmi aiutano a comprendere i solidi controlli in atto presso AWS per mantenere la sicurezza e la conformità del cloud. I report di audit di questi programmi sono disponibili per il download per i clienti AWS da AWS Artifact.

   Indipendentemente dai servizi AWS utilizzati, c'è sempre una responsabilità del cliente e le mitigazioni allineate a tale responsabilità devono essere incluse nel modello di minaccia. Per quanto riguarda le mitigazioni dei controlli di sicurezza per i servizi AWS stessi, è necessario

considerare l'implementazione dei controlli di sicurezza in tutti i domini, compresi quelli quali la gestione delle identità e degli accessi (autenticazione e autorizzazione), la protezione dei dati (a riposo e in transito), la sicurezza dell'infrastruttura, la registrazione e il monitoraggio. La documentazione di ogni servizio AWS ha un capitolo sulla sicurezza dedicato che fornisce indicazioni sui controlli di sicurezza da considerare come mitigazioni. È importante considerare il codice che si sta scrivendo e le sue dipendenze e pensare ai controlli che si possono mettere in atto per affrontare queste minacce. Questi controlli possono essere elementi come la convalida degli input, la gestione delle sessioni e la gestione dei limiti. Spesso la maggior parte delle vulnerabilità viene introdotta nel codice personalizzato, quindi è bene concentrarsi su quest'area.

4. Abbiamo fatto un buon lavoro?

   L'obiettivo è che il team e l'organizzazione migliorino sia la qualità dei modelli di minacce sia la velocità con cui vengono eseguiti nel tempo. Questi miglioramenti derivano da una combinazione di pratica, apprendimento, insegnamento e revisione. Per approfondire e mettere mano alla situazione, è consigliabile completare il corso di formazione Threat modeling the right way for builders (Come modellare le minacce nel modo giusto per gli sviluppatori) o il workshop insieme al team. Inoltre, se si desidera una guida su come integrare la modellazione delle minacce nel ciclo di vita dello sviluppo dell'applicazione della propria organizzazione, invitiamo a consultare il post How to approach threat modeling (Come affrontare la modellazione delle minacce) su AWS Security Blog (Blog sulla sicurezza AWS).

Threat Composer

Come ausilio nella modellazione delle minacce, puoi utilizzare lo strumento Threat Composer, il cui scopo è ridurre il time-to-value di questa attività. Lo strumento consente di eseguire le seguenti operazioni:

- Scrivere dichiarazioni sulle minacce in linea con la sintassi delle minacce che funzionino in un flusso di lavoro naturale non lineare
- Generare un modello di minaccia leggibile dall'uomo
- Generare un modello di minaccia leggibile dal computer per consentire la gestione dei modelli di minaccia come codice
- Velocizzare l'individuazione delle aree di miglioramento della qualità e della copertura utilizzando l'area del pannello di controllo contenente le informazioni dettagliate

Per ulteriori riferimenti, visita la pagina relativa allo strumento Threat Composer e passa all'area di lavoro di esempio definita dal sistema.

Risorse

Best practice correlate:

- SEC01-BP03 Identificazione e convalida degli obiettivi di controllo
- SEC01-BP04 Aggiornamento continuo sulle minacce alla sicurezza e sulle raccomandazioni
- SEC01-BP05 Riduzione dell'ambito di gestione della sicurezza
- SEC01-BP08 Valutazione e implementazione periodiche di nuovi servizi e funzionalità di sicurezza

Documenti correlati:

- How to approach threat modeling (AWS Security Blog)
- NIST: Guide to Data-Centric System Threat Modelling

Video correlati:

- AWS Summit ANZ 2021 – How to approach threat modelling (Summit ANZ 2021 – Come affrontare la modellazione delle minacce)
- AWS Summit ANZ 2022 - Scaling security – Optimise for fast and secure delivery (Summit ANZ 2022 - Scalare la sicurezza - Ottimizzare la consegna rapida e sicura)

Training correlati:

- Threat modeling the right way for builders – AWS Skill Builder virtual self-paced training (La corretta modellazione delle minacce per gli sviluppatori – Formazione virtuale autogestita Skill Builder)
- Threat modeling the right way for builders – AWS Workshop (La corretta modellazione delle minacce per gli sviluppatori – Workshop)

Strumenti correlati:

- Threat Composer

SEC01-BP08 Valutazione e implementazione periodiche di nuovi servizi e funzionalità di sicurezza

Valuta e implementa servizi e funzionalità di sicurezza di AWS e partner AWS che consentano di sviluppare l'assetto di sicurezza del carico di lavoro.

Risultato desiderato: hai adottato una prassi standard che ti informa sulle nuove funzionalità e servizi rilasciati da AWS e dai partner AWS. Puoi valutare come queste nuove funzionalità influenzino la progettazione di controlli attuali e nuovi per i tuoi ambienti e carichi di lavoro.

Anti-pattern comuni:

- Non devi iscriverti ai blog e ai feed RSS di AWS per conoscere rapidamente le nuove funzionalità e i servizi più importanti.
- Puoi fare affidamento su notizie e aggiornamenti sui servizi e sulle funzioni di sicurezza provenienti da fonti di seconda mano
- Non incoraggi gli utenti AWS della tua organizzazione a rimanere informati sugli ultimi aggiornamenti

Vantaggi della definizione di questa best practice: seguire i nuovi servizi e le nuove funzioni di sicurezza consente di prendere decisioni informate sull'implementazione dei controlli negli ambienti e nei carichi di lavoro cloud. Queste origini contribuiscono ad aumentare la consapevolezza dell'evoluzione del panorama della sicurezza e di come i servizi AWS possano essere utilizzati per proteggersi dalle minacce nuove ed emergenti.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

AWS informa i clienti sui nuovi servizi e funzionalità di sicurezza attraverso diversi canali:

- Novità di AWS
- AWS News Blog
- AWS Security Blog
- Bollettini sulla sicurezza di AWS
- Panoramica della documentazione AWS

Puoi iscriverti a un argomento AWS Daily Feature Updates utilizzando Amazon Simple Notification Service (Amazon SNS) per un riepilogo giornaliero completo degli aggiornamenti. Alcuni servizi di

sicurezza, come Amazon GuardDuty e AWS Security Hub, forniscono i propri argomenti SNS per rimanere informati su nuovi standard, scoperte e altri aggiornamenti per quei particolari servizi.

I nuovi servizi e le nuove funzionalità vengono inoltre annunciati e descritti in dettaglio nel corso di conferenze, eventi e webinar condotti in tutto il mondo ogni anno. Di particolare rilievo sono la conferenza annuale sulla sicurezza AWS re:Inforce e la conferenza più generale AWS re:Invent. I canali di notizie AWS menzionati in precedenza condividono questi annunci di conferenze sulla sicurezza e altri servizi e puoi guardare le sessioni didattiche approfondite online sul canale AWS Events su YouTube.

Puoi anche chiedere al tuo team di Account AWS gli ultimi aggiornamenti e consigli sui servizi di sicurezza. Puoi contattare il tuo team tramite il modulo di supporto alle vendite se non disponi delle loro informazioni di contatto diretto. Allo stesso modo, se ti sei abbonato a AWS Enterprise Support, riceverai aggiornamenti settimanali dal tuo Technical Account Manager (TAM) e potrai programmare un incontro di revisione regolare con lui.

Passaggi dell'implementazione

1. Iscriviti ai vari blog e bollettini con il tuo lettore RSS preferito o all'argomento Daily Features Updates SNS.
2. Valuta gli eventi AWS a cui partecipare per conoscere in prima persona nuove funzionalità e servizi.
3. Organizza riunioni con il team Account AWS per qualsiasi domanda sull'aggiornamento dei servizi e delle funzionalità di sicurezza.
4. Prendi in considerazione la possibilità di abbonarti a Enterprise Support per avere consulenze regolari con un Technical Account Manager (TAM).

Risorse

Best practice correlate:

- PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili
- COST01-BP07 Mantenimento dell'aggiornamento sulle nuove versioni dei servizi

Gestione di identità e accessi

Domande

- SEC 2. Come si gestisce l'autenticazione per persone e macchine?
- SEC 3. Come si gestisce l'autenticazione per persone e macchine?

SEC 2. Come si gestisce l'autenticazione per persone e macchine?

Ci sono due tipi di identità da gestire quando inizi a utilizzare carichi di lavoro AWS sicuri. Comprendere il tipo di identità necessaria per gestire e concedere l'accesso ti aiuta a verificare che le identità corrette abbiano accesso alle risorse giuste nelle condizioni adeguate.

Identità umane: amministratori, sviluppatori, operatori e utenti finali necessitano di un'identità per accedere agli ambienti e alle applicazioni AWS. Si tratta di membri dell'organizzazione o di utenti esterni con cui collabori e che interagiscono con le risorse AWS tramite Web browser, applicazioni client o strumenti a riga di comando interattivi.

Identità di macchine: le applicazioni di servizio, gli strumenti operativi e i carichi di lavoro necessitano di un'identità per effettuare richieste ai servizi AWS, ad esempio per leggere i dati. Queste identità includono macchine in esecuzione nell'ambiente AWS, ad esempio istanze Amazon EC2 o funzioni AWS Lambda. Puoi gestire le identità di macchine anche per soggetti esterni che necessitano dell'accesso. Inoltre, potresti disporre di macchine al di fuori di AWS che devono accedere al tuo ambiente AWS.

Best practice

- SEC02-BP01 Utilizzo meccanismi di accesso efficaci
- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC02-BP03 Archiviazione e utilizzo dei segreti in modo sicuro
- SEC02-BP04 Fai affidamento su un provider di identità centralizzato
- SEC02-BP05 Verifica e rotazione periodica delle credenziali
- SEC02-BP06 Impiego dei gruppi di utenti e degli attributi

SEC02-BP01 Utilizzo meccanismi di accesso efficaci

I sign-in (autenticazione tramite credenziali di accesso) possono presentare dei rischi se non si utilizzano meccanismi come l'autenticazione a più fattori (MFA), soprattutto in situazioni in cui le credenziali di accesso sono state inavvertitamente divulgate o sono facilmente identificabili. Utilizza meccanismi di accesso efficaci per ridurre questi rischi, richiedendo l'MFA e policy sulle password sicure.

Risultato desiderato: ridurre i rischi di accesso involontario alle credenziali in AWS utilizzando meccanismi di accesso efficaci per gli utenti AWS Identity and Access Management (IAM), l'utente root Account AWS, AWS IAM Identity Center (successore di AWS Single Sign-On) e i provider di identità di terze parti. Ciò significa richiedere l'MFA, applicare policy sulle password efficaci e rilevare comportamenti di accesso anomali.

Anti-pattern comuni:

- Nessuna applicazione di policy sulle password efficaci per le proprie identità, comprese password complesse e MFA.
- Condivisione delle stesse credenziali tra utenti diversi.
- Nessun utilizzo di controlli investigativi per gli accessi sospetti.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Ci sono molti modi in cui le identità umane possono accedere ad AWS. È una best practice di AWS affidarsi a un provider di identità centralizzato che si avvale della federazione (federazione diretta o utilizzo di AWS IAM Identity Center) per l'autenticazione ad AWS. In questo caso, è necessario stabilire un processo di accesso sicuro con il provider di identità o con Microsoft Active Directory.

Quando apri un Account AWS, inizi con un utente root Account AWS. L'account utente root deve essere utilizzato solo per impostare l'accesso per gli utenti e per le attività che richiedono l'utente root). È importante abilitare l'MFA per l'utente root dell'account subito dopo l'apertura di Account AWS e proteggere l'utente root usando la guida AWS alle best practice.

Se crei utenti in AWS IAM Identity Center, proteggi il processo di accesso in quel servizio. Per le identità dei consumatori, puoi usare Amazon Cognito user pools e proteggere il processo di accesso in tale servizio oppure puoi utilizzare uno dei fornitori di identità supportato da Amazon Cognito user pools.

Se si utilizzano gli utenti AWS Identity and Access Management (IAM), è opportuno proteggere il processo di accesso mediante IAM.

Indipendentemente dal metodo di accesso, è fondamentale applicare una policy di accesso efficace.

Passaggi dell'implementazione

Le seguenti sono raccomandazioni generali per l'accesso sicuro. Le impostazioni effettive da configurare devono essere stabilite dalla policy aziendale o utilizzare uno standard come NIST 800-63.

- Richiedere l'MFA. Richiedere l'MFA è una best practice IAM per le identità e i carichi di lavoro umani. L'abilitazione dell'MFA fornisce un ulteriore livello di sicurezza che richiede agli utenti di fornire le credenziali di accesso e un codice OTP (One-Time Password) o una stringa verificata e generata crittograficamente da un dispositivo hardware.
- Applicare una lunghezza minima della password, che è un fattore primario nella forza della password.
- Applicare la complessità delle password in modo che sia più difficile individuarle.
- Consentire agli utenti di modificare le proprie password.
- Creare identità individuali invece di credenziali condivise. Creando identità individuali, è possibile assegnare a ciascun utente un set unico di credenziali di sicurezza. I singoli utenti consentono di sottoporre a audit l'attività di ciascuno.

Suggerimenti IAM Identity Center:

- IAM Identity Center fornisce una policy sulla password prestabilita quando si utilizza la directory predefinita che stabilisce i requisiti di lunghezza, complessità e riutilizzo delle password.
- Abilitare l'MFA e configurare l'impostazione "Compatibile con il contesto" o "Sempre attivo" per l'MFA quando l'origine dell'identità è la directory predefinita, AWS Managed Microsoft AD o AD Connector.
- Consenti agli utenti di registrare i propri dispositivi MFA.

Suggerimenti sulla directory Amazon Cognito user pools:

- Configura le impostazioni di forza della password.
- Richiedi l'MFA per gli utenti.
- Utilizza le Amazon Cognito user pools impostazioni di sicurezza avanzate per le funzionalità quali l'autenticazione adattiva che può bloccare sign-in sospetti.

Suggerimenti per l'utente IAM:

- Idealmente stai utilizzando IAM Identity Center o la federazione diretta. Tuttavia, potrebbero essere necessari utenti IAM. In tal caso, imposta una policy sulla password per gli utenti IAM. Puoi utilizzare la policy sulla password per definire requisiti quali la lunghezza minima o la necessità che la password richieda caratteri non alfabetici.
- Crea una policy IAM per applicare l'accesso MFA in modo che gli utenti possano gestire le proprie password e i dispositivi MFA.

Risorse

Best practice correlate:

- SEC02-BP03 Archiviazione e utilizzo dei segreti in modo sicuro
- SEC02-BP04 Fai affidamento su un provider di identità centralizzato
- SEC03-BP08 Condivisione delle risorse in modo sicuro all'interno dell'organizzazione

Documenti correlati:

- AWS IAM Identity Center (successor to AWS Single Sign-On) Password Policy Policy sulle password AWS IAM Identity Center (successore di AWS Single Sign-On)
- IAM user password policy (Policy sulle password degli utenti IAM)
- Setting the Account AWS root user password (Impostazione della password dell'utente root dell'account AWS)
- Amazon Cognito password policy (Policy sulla password di Amazon Cognito)
- AWS credentials (Credenziali AWS)
- IAM security best practices (Best Practice di sicurezza IAM)

Video correlati:

- Managing user permissions at scale with AWS IAM Identity Center (Gestire le autorizzazioni degli utenti su larga scala con AWS SSO)
- Mastering identity at every layer of the cake

SEC02-BP02 Utilizzo di credenziali temporanee

Quando si esegue qualsiasi tipo di autenticazione, è preferibile utilizzare credenziali temporanee invece di credenziali a lungo termine per ridurre o eliminare i rischi, come la divulgazione, la condivisione o il furto involontario delle credenziali.

Risultato desiderato: per ridurre il rischio legato alle credenziali a lungo termine, utilizza credenziali temporanee ogni qualvolta sia possibile sia per le identità umane che per le identità macchina. Le credenziali a lungo termine creano molti rischi, ad esempio possono essere caricate in codice su repository GitHub pubblici. Utilizzando credenziali temporanee, riduci notevolmente le possibilità di compromissione delle credenziali.

Anti-pattern comuni:

- Sviluppatori che utilizzano chiavi di accesso a lungo termine dagli IAM users anziché ottenere credenziali temporanee dalla CLI utilizzando la federazione.
- Sviluppatori che inseriscono chiavi di accesso a lungo termine nel loro codice e caricano tale codice su repository Git pubblici.
- Sviluppatori che inseriscono chiavi di accesso a lungo termine nelle applicazioni mobili che vengono poi rese disponibili negli app store.
- Utenti che condividono le chiavi di accesso a lungo termine con altri utenti o dipendenti che lasciano l'azienda con chiavi di accesso a lungo termine ancora in loro possesso.
- Utilizzo di chiavi di accesso a lungo termine per le identità macchina quando è possibile utilizzare credenziali temporanee.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Utilizza credenziali di sicurezza temporanee invece di credenziali a lungo termine per tutte le richieste API e CLI AWS. Le richieste API e CLI ai servizi AWS devono, in quasi tutti i casi, essere firmate utilizzando le chiavi di accesso AWS. Queste richieste possono essere firmate con credenziali temporanee o a lungo termine. L'unico caso in cui si devono utilizzare credenziali a lungo termine, note anche come chiavi di accesso a lungo termine, è qualora si stia utilizzando un utente IAM o un utente root Account AWS. Al momento della federazione ad AWS o dell'assunzione di un ruolo IAM attraverso altri metodi, vengono generate delle credenziali temporanee. Anche quando accedi a AWS Management Console utilizzando le credenziali di accesso, vengono generate credenziali

temporanee per effettuare chiamate ai servizi AWS. Sono poche le situazioni in cui è necessario disporre di credenziali a lungo termine ed è possibile svolgere quasi tutte le attività utilizzando credenziali temporanee.

Evitare l'uso di credenziali a lungo termine a favore di credenziali temporanee dovrebbe andare di pari passo con una strategia di riduzione dell'uso degli utenti IAM a favore della federazione e dei ruoli IAM. Sebbene in passato gli utenti IAM siano stati utilizzati sia per le identità umane che per quelle macchina, ora si consiglia di non utilizzarli per evitare i rischi legati all'uso di chiavi di accesso a lungo termine.

Passaggi dell'implementazione

Per le identità umane come dipendenti, amministratori, sviluppatori, operatori e clienti:

- Devi affidarti a un fornitore di identità centralizzato e richiedere agli utenti umani di utilizzare la federazione con un fornitore di identità per accedere ad AWS utilizzando credenziali temporanee. La federazione degli utenti può essere effettuata con la federazione diretta a ciascun Account AWS o utilizzando AWS IAM Identity Center (successore di AWS IAM Identity Center) e un provider di identità a scelta. La federazione offre una serie di vantaggi rispetto all'utilizzo degli utenti IAM, oltre all'eliminazione delle credenziali a lungo termine. Gli utenti possono anche richiedere credenziali temporanee dalla riga di comando per la federazione diretta o utilizzare IAM Identity Center. Ciò significa che i casi d'uso che richiedono utenti IAM o credenziali a lungo termine per gli utenti sono pochi.
- Quando concedi a terzi, come ad esempio ai fornitori di software come servizio (SaaS), l'accesso alle risorse del tuo Account AWS, puoi utilizzare ruoli multi-account e policy basate sulle risorse.
- Se devi concedere l'accesso alle tue risorse alle applicazioni per i consumatori o per i clientiAWS, puoi utilizzare i pool di identità Amazon Cognito o Amazon Cognito user pools per fornire le credenziali temporanee. Le autorizzazioni per le credenziali sono configurate tramite i ruoli IAM. Puoi anche definire un ruolo IAM separato con autorizzazioni limitate per gli utenti guest non autenticati.

Per le identità macchina, potrebbero essere necessarie credenziali a lungo termine. In questi casi, devi richiedere ai carichi di lavoro di utilizzare credenziali temporanee con ruoli IAM per accedere ad AWS.

- Per Amazon Elastic Compute Cloud (Amazon EC2), puoi utilizzare ruoli per Amazon EC2.

- AWS Lambda ti consente di configurare un ruolo di esecuzione Lambda per concedere le autorizzazioni al servizio per eseguire azioni AWS utilizzando credenziali temporanee. Per i servizi AWS esistono molti altri modelli simili per concedere credenziali temporanee utilizzando i ruoli IAM.
- Per i dispositivi IoT, puoi utilizzare il provider di credenziali AWS IoT Core per richiedere credenziali temporanee.
- Per i sistemi on-premise o per i sistemi che vengono eseguiti al di fuori di AWS che richiedono accesso alle risorse AWS, puoi utilizzare IAM Roles Anywhere.

Esistono scenari in cui le credenziali temporanee non sono un'opzione e potrebbe essere necessario utilizzare credenziali a lungo termine. In queste situazioni, sottoponi a audit e ruota periodicamente le credenziali e ruota regolarmente le chiavi di accesso per i casi d'uso che richiedono credenziali a lungo termine.. Alcuni esempi che potrebbero richiedere credenziali a lungo termine sono i plugin di WordPress e i client AWS di terze parti. Quando è necessario utilizzare credenziali a lungo termine o per credenziali diverse dalle chiavi di accesso AWS, come ad esempio i login ai database, puoi utilizzare un servizio progettato per gestire i segreti, ad esempio AWS Secrets Manager. Secrets Manager consente di gestire, ruotare e archiviare in modo semplice e sicuro i segreti crittografati usando servizi supportati. Per ulteriori informazioni sulla rotazione delle credenziali a lungo termine, consulta Rotating Access Keys (Rotazione delle chiavi di accesso).

Risorse

Best practice correlate:

- SEC02-BP03 Archiviazione e utilizzo dei segreti in modo sicuro
- SEC02-BP04 Fai affidamento su un provider di identità centralizzato
- SEC03-BP08 Condivisione delle risorse in modo sicuro all'interno dell'organizzazione

Documenti correlati:

- Credenziali di sicurezza temporanee
- AWS Credentials (Credenziali AWS)
- IAM Security Best Practices (Best practice per la sicurezza IAM)
- Ruoli IAM
- IAM Identity Center
- Provider di identità e federazione

- Rotating Access Keys
- Soluzioni dei partner per la sicurezza: accesso e controllo degli accessi
- L'utente root dell'account AWS

Video correlati:

- Managing user permissions at scale with AWS IAM Identity Center (Gestire le autorizzazioni degli utenti su larga scala con AWS SSO), successore di AWS IAM Identity Center)
- Mastering identity at every layer of the cake

SEC02-BP03 Archiviazione e utilizzo dei segreti in modo sicuro

Un carico di lavoro richiede una capacità automatizzata di dimostrare la propria identità a database, risorse e servizi di terze parti. A tal fine si utilizzano credenziali di accesso segrete, come chiavi di accesso API, password e token OAuth. L'utilizzo di un servizio appositamente creato per archiviare, gestire e ruotare queste credenziali aiuta a ridurre la probabilità che queste vengano compromesse.

Risultato desiderato: implementare un meccanismo per la gestione sicura delle credenziali delle applicazioni che raggiunga i seguenti obiettivi:

- Identificare i segreti necessari per il carico di lavoro.
- Ridurre il numero di credenziali a lungo termine sostituendole con credenziali a breve termine, quando possibile.
- Stabilire l'archiviazione sicura e la rotazione automatica delle rimanenti credenziali a lungo termine.
- Sottoporre a audit l'accesso ai segreti esistenti nel carico di lavoro.
- Eseguire il monitoraggio continuo per verificare che nessun segreto sia incorporato nel codice sorgente durante il processo di sviluppo.
- Ridurre la probabilità che le credenziali vengano divulgate inavvertitamente.

Anti-pattern comuni:

- Nessuna rotazione delle credenziali.
- Memorizzazione di credenziali a lungo termine nel codice sorgente o nei file di configurazione.
- Memorizzazione delle credenziali a riposo non criptate.

Vantaggi dell'adozione di questa best practice:

- I segreti sono conservati in modo criptato a riposo e in transito.
- L'accesso alle credenziali è regolato da un'API (si pensi a un distributore automatico di credenziali).
- L'accesso a una credenziale (sia in lettura che in scrittura) viene sottoposto a audit e registrato.
- Separazione delle preoccupazioni: la rotazione delle credenziali viene eseguita da un componente distinto, che può essere separato dal resto dell'architettura.
- I segreti vengono distribuiti automaticamente su richiesta ai componenti software e la rotazione avviene in una posizione centrale.
- L'accesso alle credenziali può essere controllato in modo granulare.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

In passato, le credenziali utilizzate per l'autenticazione ai database, alle API di terze parti, ai token e ad altri segreti potevano essere incorporate nel codice sorgente o nei file di ambiente. AWS fornisce diversi meccanismi per memorizzare queste credenziali in modo sicuro, ruotarle automaticamente e sottoporre a audit il loro utilizzo.

Il modo migliore per affrontare la gestione dei segreti è seguire le indicazioni di rimuovere, sostituire e ruotare. La credenziale più sicura è quella che non si deve memorizzare, gestire o trattare. Possono esserci credenziali che non sono più necessarie per il funzionamento del carico di lavoro e che possono essere rimosse in modo sicuro.

Per le credenziali che sono ancora necessarie per il corretto funzionamento del carico di lavoro, potrebbe esserci l'opportunità di sostituire una credenziale a lungo termine con una credenziale temporanea o a breve termine. Ad esempio, invece di una codifica fissa di una chiave di accesso segreta AWS, si può pensare di sostituire la credenziale a lungo termine con una credenziale temporanea utilizzando i ruoli IAM.

Alcuni segreti di lunga durata potrebbero non poter essere rimossi o sostituiti. Questi segreti possono essere memorizzati in un servizio come AWS Secrets Manager, dove possono essere archiviati, gestiti e ruotati regolarmente a livello centrale.

Un audit del codice sorgente e dei file di configurazione del carico di lavoro può rivelare molti tipi di credenziali. La tabella seguente riassume le strategie per gestire i tipi più comuni di credenziali:

Credential type	Description	Suggested strategy
IAM access keys	AWS IAM access and secret keys used to assume IAM roles inside of a workload	Replace: Use Ruoli IAM assigned to the compute instances (such as Amazon EC2 or AWS Lambda) instead. For interoperability with third parties that require access to resources in your Account AWS, ask if they support AWS cross-account access (Accesso multi-account AWS). For mobile apps, consider using temporary credentia ls through Pool di identità di Amazon Cognito (identità federate). For workloads running outside of AWS, consider IAM Roles Anywhere or Attivazioni ibride AWS Systems Manager.
SSH keys	Secure Shell private keys used to log into Linux EC2 instances, manually or as part of an automated process	Replace: Use AWS Systems Manager or EC2 Instance Connect to provide programmatic and human access to EC2 instances using IAM roles.
Application and database credentials	Passwords – plain text string	Rotate: Store credentials in AWS Secrets Manager and establish automated rotation if possible.
Amazon RDS and Aurora Admin Database credentials	Passwords – plain text string	Replace: Use the Secrets Manager integration with Amazon RDS (Integrazione di

Credential type	Description	Suggested strategy
		Secrets Manager con Amazon RDS) or Amazon Aurora. In addition, some RDS database types can use IAM roles instead of passwords for some use cases (for more detail, see IAM database authentication (Autenticazione del database IAM)).
OAuth tokens	Secret tokens – plain text string	Rotate: Store tokens in AWS Secrets Manager and configure automated rotation.
API tokens and keys	Secret tokens – plain text string	Rotate: Store in AWS Secrets Manager and establish automated rotation if possible.

Un anti-pattern comune è quello di incorporare le chiavi di accesso IAM all'interno del codice sorgente, dei file di configurazione o delle applicazioni mobili. Quando è richiesta una chiave di accesso IAM per comunicare con un servizio AWS, utilizza le credenziali di sicurezza temporanee (a breve termine). Queste credenziali a breve termine possono essere fornite attraverso ruoli IAM per istanze EC2, ruoli di esecuzione per funzioni Lambda, ruoli Cognito IAM per l'accesso degli utenti di dispositivi mobili e policy IoT Core per i dispositivi IoT. Quando ci si interfaccia con terze parti, è preferibile delegare l'accesso a un ruolo IAM con l'accesso necessario alle risorse del proprio account, piuttosto che configurare un utente IAM e inviare alla terza parte la chiave di accesso segreta per quell'utente.

In molti casi il carico di lavoro richiede la memorizzazione di segreti necessari per l'interoperabilità con altri servizi e risorse. AWS Secrets Manager è costruito appositamente per gestire in modo sicuro queste credenziali, nonché l'archiviazione, l'uso e la rotazione di token API, password e altre credenziali.

AWS Secrets Manager fornisce cinque funzionalità chiave per garantire l'archiviazione e la gestione sicura delle credenziali sensibili: crittografia a riposo, crittografia in transito, audit completo, controllo degli accessi granulare e rotazione estensibile delle credenziali. Sono accettabili anche altri servizi di

gestione dei segreti dei partner AWS o soluzioni sviluppate localmente che forniscano funzionalità e garanzie simili.

Passaggi dell'implementazione

1. Individuare i percorsi di codice contenenti credenziali hard-coded utilizzando strumenti automatizzati come Amazon CodeGuru.
   - Utilizzare Amazon CodeGuru per eseguire la scansione dei repository di codice. Una volta completata la revisione, filtrare `Type=Secrets` in CodeGuru per trovare le linee di codice problematiche.
2. Identificare le credenziali che possono essere rimosse o sostituite.
   a. Identificare le credenziali non più necessarie e contrassegnarle per la rimozione.
   b. Le chiavi segrete AWS incorporate nel codice sorgente devono essere sostituite con ruoli IAM associati alle risorse necessarie. Se una parte del carico di lavoro è al di fuori di AWS ma richiede le credenziali IAM per accedere alle risorse AWS, considerare IAM Roles Anywhere o Attivazioni ibride AWS Systems Manager.
3. Per altri segreti di terze parti a lunga durata che richiedono l'uso della strategia di rotazione, integrare Secrets Manager nel codice per recuperare i segreti di terze parti in fase di esecuzione.
   a. La console CodeGuru può creare un segreto in Secrets Manager automaticamente utilizzando le credenziali individuate.
   b. Integrare il recupero dei segreti da Secrets Manager nel codice dell'applicazione.
      - Le funzioni Lambda serverless possono utilizzare un'estensione Lambda indipendente dal linguaggio.
      - Per le istanze o i container EC2, AWS fornisce esempi di codice lato client per il recupero dei segreti da Secrets Manager in diversi linguaggi di programmazione popolari.
4. Esaminare periodicamente la base di codice e ripetere la scansione per verificare che non siano stati aggiunti nuovi segreti al codice.
   - Valutare l'utilizzo di uno strumento come git-secrets per impedire il commit di nuovi segreti nel repository del codice sorgente.
5. Monitorare l'attività di Secrets Manager per rilevare indicazioni di utilizzo inatteso, accesso inappropriato ai segreti o tentativi di cancellazione dei segreti.
6. Ridurre l'esposizione umana alle credenziali. Limitare l'accesso alle credenziali di lettura, scrittura e modifica a un ruolo IAM dedicato a questo scopo e fornire l'accesso per assumere il ruolo solo a un piccolo sottoinsieme di utenti operativi.

Risorse

Best practice correlate:

- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC02-BP05 Verifica e rotazione periodica delle credenziali

Documenti correlati:

- Nozioni di base su AWS Secrets Manager
- Provider di identità e federazione
- Amazon CodeGuru Introduces Secrets Detector (Amazon CodeGuru introduce il rivelatore di segreti)
- How AWS Secrets Manager uses AWS Key Management Service(In che modo AWS Secrets Manager utilizza AWS Key Management Service)
- Crittografia e decrittografia del segreto in Secrets Manager
- Articoli del blog su Secrets Manager
- Amazon RDS announces integration with AWS Secrets Manager (Amazon RDS announces integration with AWS Secrets Manager)

Video correlati:

- Best practice per la gestione, il recupero e la rotazione di segreti su scala
- Find Hard-Coded Secrets Using Amazon CodeGuru Secrets Detector (Trovare i segreti codificati usando il rilevatore di segreti di Amazon CodeGuru)
- Securing Secrets for Hybrid Workloads Using AWS Secrets Manager (Trovare i segreti codificati usando il rilevatore di segreti di Amazon CodeGuru)

Workshop correlati:

- Store, retrieve, and manage sensitive credentials in AWS Secrets Manager (Memorizzare, recuperare e gestire le credenziali sensibili in AWS Secrets Manager)
- Attivazioni ibride AWS Systems Manager

SEC02-BP04 Fai affidamento su un provider di identità centralizzato

Per le identità della forza lavoro (dipendenti e collaboratori) affidati a un provider di identità digitale che ti consenta di gestire le identità in un luogo centralizzato. In questo modo è più semplice gestire l'accesso tra più applicazioni e sistemi, perché crei, assegni, gestisci, revochi e verifichi gli accessi da una singola posizione.

Risultato desiderato: Hai un provider di identità centralizzato dal quale gestisci centralmente gli utenti della forza lavoro, le policy di autenticazione (come le richieste di autenticazione a più fattori o MFA) e le autorizzazioni per sistemi e applicazioni, come l'assegnazione dell'accesso in base all'appartenenza o agli attributi di un utente. Gli utenti che fanno parte della tua forza lavoro accedono al provider di identità centrale ed effettuano l'accesso federato (autenticazione unica) alle applicazioni interne ed esterne, il che elimina la necessità per gli utenti di ricordare più credenziali. Il provider di identità è integrato con i tuoi sistemi di risorse umane (HR), in modo che le modifiche relative al personale vengano sincronizzate automaticamente con il provider di identità. Ad esempio, se qualcuno lascia l'organizzazione, puoi revocare automaticamente l'accesso alle applicazioni e ai sistemi federati (incluso AWS). Hai abilitato la verifica dettagliata dei log nel tuo provider di identità e stai monitorando questi log per rilevare comportamenti degli utenti insoliti.

Anti-pattern comuni:

- Non utilizzi la federazione e l'autenticazione unica. Gli utenti che appartengono alla tua forza lavoro creano account utente e credenziali separati in più applicazioni e sistemi.
- Non hai automatizzato il ciclo di vita delle identità degli utenti che fanno parte della tua forza lavoro, ad esempio integrando il provider di identità con i tuoi sistemi HR. Quando un utente lascia l'organizzazione o cambia ruolo, segui una procedura manuale per eliminare o aggiornare i suoi record in più applicazioni e sistemi.

Vantaggi dell'adozione di questa best practice: Utilizzare un provider di identità centralizzato ti fornisce un unico posto per gestire le identità e le policy degli utenti che fanno parte della tua forza lavoro, la possibilità di assegnare l'accesso alle applicazioni a utenti e gruppi e la possibilità di monitorare l'attività di accesso degli utenti. Grazie all'integrazione con i sistemi di risorse umane (HR), quando un utente cambia ruolo, queste modifiche vengono sincronizzate con il provider di identità e le applicazioni e le autorizzazioni assegnate vengono aggiornate automaticamente. Quando un utente lascia l'organizzazione, la sua identità viene automaticamente disabilitata nel provider di identità e l'accesso alle applicazioni e ai sistemi federati viene revocato.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Linee guida per l'accesso ad AWS degli utenti che fanno parte della forza lavoro

Gli utenti che fanno parte della forza lavoro, come dipendenti e collaboratori dell'organizzazione, potrebbero richiedere l'accesso ad AWS per utilizzare la AWS Management Console o la AWS Command Line Interface (AWS CLI) per svolgere le proprie mansioni lavorative. Puoi concedere l'accesso ad AWS a tali utenti federando il tuo provider di identità centralizzato AWS a due livelli: federazione diretta a ciascun Account AWS o federazione a più account della tua organizzazione AWS.

- Per federare gli utenti della tua forza lavoro direttamente con ciascuno Account AWS, utilizza un provider di identità centralizzato per federare l'accesso a AWS Identity and Access Management in quell'account. Grazie alla sua flessibilità, IAM ti permette di abilitare un SAML 2.0 o un Open ID Connect (OIDC) Identity Provider per ciascun Account AWS e di utilizzare attributi utente federati per il controllo degli accessi. Gli utenti della tua forza lavoro utilizzano il proprio browser Web per accedere al provider di identità e forniscono le proprie credenziali (come password e codici token MFA). Il provider di identità rilascia un'asserzione SAML nel browser che viene inviata all'URL di accesso della AWS Management Console, così da consentire all'utente di accedere mediante l'autenticazione unica alla AWS Management Console tramite l'assunzione di un ruolo IAM. Gli utenti possono anche ottenere credenziali API AWS temporanee da utilizzare nella AWS CLI o AWS SDK da AWS STS tramite l'assunzione del ruolo IAM utilizzando un'asserzione SAML ottenuta dal provider di identità.
- Per federare gli utenti della tua forza lavoro con più account all'interno dell'organizzazione AWS, puoi usare AWS IAM Identity Center per gestire centralmente l'accesso degli utenti agli Account AWS e alle applicazioni. Attiva Centro di identità per la tua organizzazione e configura la tua origine di identità. IAM Identity Center fornisce una directory di origine delle identità predefinita, che puoi utilizzare per gestire utenti e gruppi. In alternativa, puoi scegliere un'origine di identità esterna connettendoti al tuo provider di identità esterno tramite SAML 2.0 ed effettuando il provisioning automatico di utenti e gruppi che utilizzano SCIM, oppure connettendoti a Microsoft AD Directory utilizzando AWS Directory Service. Una volta configurata un'origine di identità, puoi assegnare l'accesso agli Account AWS a utenti e gruppi, definendo policy di privilegio minimo nel tuo set di autorizzazioni. Gli utenti della tua forza lavoro possono autenticarsi tramite il provider di identità centrale per accedere al portale di accesso AWS ed effettuare l'autenticazione unica per ottenere l'accesso agli Account AWS e alle applicazioni cloud a loro assegnate. Gli utenti possono configurare AWS CLI v2 per autenticarsi con Centro di identità e ottenere le credenziali per eseguire comandi della AWS CLI. Centro di identità consente inoltre l'accesso tramite

autenticazione unica ad applicazioni AWS come Amazon SageMaker Studio e ai portali AWS IoT Sitewise Monitor.

Dopo aver seguito le indicazioni precedenti, gli utenti della forza lavoro non avranno più bisogno di utilizzare IAM users e gruppi per le normali operazioni quando gestiscono i carichi di lavoro su AWS. Al contrario, gli utenti e i gruppi sono gestiti all'esterno di AWS e gli utenti possono accedere alle risorse AWS come identità federata. Le identità federate utilizzano i gruppi definiti dal provider di identità centralizzato. Devi identificare e rimuovere i gruppi IAM, gli IAM users e le credenziali utente di lunga durata (password e chiavi di accesso) che non sono più necessarie nei tuoi Account AWS. Puoi trovare credenziali inutilizzate utilizzando il report sulle credenziali IAM, eliminare gli IAM users interessati e rimuovere i gruppi IAM. Puoi applicare una policy di controllo dei servizi (SCP) alla tua organizzazione per prevenire la creazione di nuovi IAM users e gruppi, applicando l'accesso ad AWS tramite identità federate.

Linee guida per gli utenti delle tue applicazioni

Puoi gestire le identità degli utenti delle applicazioni, ad esempio un'app per dispositivi mobili, utilizzando Amazon Cognito come provider di identità centralizzato. Amazon Cognito consente l'autenticazione, l'autorizzazione e la gestione degli utenti per le app Web e mobili. Amazon Cognito fornisce un archivio di identità dimensionabile fino a milioni di utenti, supporta la federazione delle identità sociali e aziendali e offre funzionalità di sicurezza avanzate per proteggere gli utenti e l'azienda. Puoi integrare la tua applicazione Web o mobile personalizzata con Amazon Cognito per aggiungere l'autenticazione degli utenti e il controllo degli accessi alle applicazioni in pochi minuti. Amazon Cognito si fonda su standard di identità aperti come SAML e Open ID Connect (OIDC), supporta varie normative di conformità e si integra con le risorse di sviluppo frontend e backend.

Passaggi dell'implementazione

Procedure per l'accesso ad AWS degli utenti che fanno parte della forza lavoro

- Federa l'accesso ad AWS degli utenti della tua forza lavoro tramite un provider di identità centralizzato seguendo uno dei seguenti approcci:
  - Utilizza IAM Identity Center per abilitare l'autenticazione unica negli Account AWS per più utenti della tua organizzazione AWS tramite la federazione con il provider di identità.
  - Utilizza IAM per connettere il provider di identità direttamente a ciascun Account AWS, abilitando un accesso federato e granulare.
- Identifica e rimuovi gli IAM users e i gruppi che vengono sostituiti da identità federate.

Passaggi per gli utenti delle tue applicazioni

- Utilizza Amazon Cognito come provider di identità centralizzato per le tue applicazioni.
- Integra le applicazioni personalizzate con Amazon Cognito utilizzando OpenID Connect e OAuth. Puoi sviluppare applicazioni personalizzate utilizzando le librerie Amplify, che forniscono interfacce semplici da integrare con una varietà di servizi AWS per l'autenticazione, come Amazon Cognito.

Risorse

Best practice Well-Architected correlate:

- SEC02-BP06 Impiego dei gruppi di utenti e degli attributi
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- SEC03-BP06 Gestione degli accessi in base al ciclo di vita

Documenti correlati:

- Identity federation in AWS
- Best practice per la sicurezza in IAM
- AWS Identity and Access Management Best practices
- Getting started with IAM Identity Center delegated administration
- How to use customer managed policies in IAM Identity Center for advanced use cases
- AWS CLI v2: IAM Identity Center credential provider

Video correlati:

- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive
- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center
- AWS re:Invent 2018: Mastering Identity at Every Layer of the Cake

Esempi correlati:

- Workshop: Using AWS IAM Identity Center to achieve strong identity management
- Workshop: Serverless identity

Strumenti correlati:

- Partner AWS con competenze nella sicurezza: gestione di identità e accessi
- saml2aws

SEC02-BP05 Verifica e rotazione periodica delle credenziali

Sottoponi a audit e ruota periodicamente le credenziali per limitarne il tempo di utilizzo per accedere alle risorse. Le credenziali a lungo termine espongono a molti rischi che possono essere ridotti ruotandole regolarmente.

Risultato desiderato: implementare la rotazione delle credenziali per ridurre i rischi associati all'utilizzo a lungo termine. Esegui regolarmente l'audit e rimedia alla non conformità con le policy di rotazione delle credenziali.

Anti-pattern comuni:

- Nessun audit dell'uso delle credenziali.
- Utilizzo non necessario di credenziali a lungo termine.
- Utilizzo di credenziali a lungo termine e mancata rotazione regolare.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Quando non si può fare affidamento sulle credenziali temporanee e sono necessarie credenziali a lungo termine, sottoponile a audit per assicurarti che siano applicati i controlli prestabiliti, ad esempio l'autenticazione a più fattori (MFA), che siano soggette a regolare rotazione e dispongano di un livello di accesso appropriato.

La convalida periodica, preferibilmente tramite uno strumento automatizzato, è necessaria per verificare che vengano applicati i controlli corretti. Per le identità umane, è necessario richiedere agli utenti di modificare periodicamente le password e ritirare le chiavi di accesso a favore delle credenziali temporanee. Quando passi da utenti AWS Identity and Access Management (IAM) a identità centralizzate, puoi generare un report delle credenziali per effettuare l'audit degli utenti.

Ti consigliamo inoltre di monitorare l'MFA nel tuo provider di identità. Puoi configurare Regole di AWS Config o utilizzare gli standard di sicurezza AWS Security Hub per verificare se gli utenti hanno l'MFA abilitata. Considera la possibilità di utilizzare IAM Roles Anywhere per fornire credenziali temporanee

per le identità macchina. Nelle situazioni in cui l'utilizzo di credenziali temporanee e ruoli IAM non è possibile, è necessario un audit frequente e la rotazione delle chiavi di accesso.

Passaggi dell'implementazione

- Eseguire regolarmente l'audit delle credenziali: l'audit delle identità configurate nel provider di identità e in IAM aiuta a verificare che solo le identità autorizzate abbiano accesso al carico di lavoro. Tali identità possono includere, a titolo esemplificativo ma non esaustivo, utenti IAM, utenti AWS IAM Identity Center, utenti Active Directory o utenti in un diverso provider di identità a monte. Ad esempio, eliminare le persone che lasciano l'organizzazione e i ruoli multi-account che non sono più necessari. Disporre di un processo per sottoporre periodicamente a audit le autorizzazioni ai servizi a cui accede un'entità IAM. Questo aiuta a identificare le policy da modificare per rimuovere le autorizzazioni non utilizzate. Utilizza i report delle credenziali e AWS Identity and Access Management Access Analyzer per eseguire l'audit di autorizzazioni e credenziali IAM. Puoi utilizzare Amazon CloudWatch per configurare allarmi per chiamate API specifiche effettuate nell'ambiente AWS. Amazon GuardDuty può anche avvisare di attività impreviste, che potrebbero indicare un accesso estremamente permissivo o un accesso non intenzionale alle credenziali IAM.
- Ruota regolarmente le credenziali: quando non è possibile utilizzare le credenziali temporanee, ruotare regolarmente le chiavi di accesso IAM a lungo termine, al massimo ogni 90 giorni. Se una chiave di accesso viene involontariamente divulgata a propria insaputa, questo limita la durata di utilizzo delle credenziali per accedere alle risorse. Per informazioni sulla rotazione delle chiavi di accesso per gli utenti IAM, consulta Rotating access keys.
- Rivedi le autorizzazioni IAM: per migliorare la sicurezza dell'Account AWS, rivedere e monitorare regolarmente ogni policy IAM. Verifica che le policy rispettino il principio del privilegio minimo.
- Considera la possibilità di automatizzare la creazione e gli aggiornamenti delle risorse IAM: IAM Identity Center automatizza molte attività IAM, come la gestione dei ruoli e delle policy. In alternativa, AWS CloudFormation può essere utilizzato per automatizzare l'implementazione delle risorse IAM, compresi ruoli e policy, per ridurre la possibilità di errore umano, poiché i modelli possono essere verificati e controllati in versione.
- Utilizza IAM Roles Anywhere per sostituire gli utenti IAM per le identità macchina: IAM Roles Anywhere consente di utilizzare i ruoli in aree tradizionalmente non accessibili, come i server on-premise. IAM Roles Anywhere utilizza un certificato X.509 affidabile per autenticarsi ad AWS e ricevere credenziali temporanee. L'utilizzo di IAM Roles Anywhere evita la necessità di ruotare queste credenziali, poiché le credenziali a lungo termine non vengono più memorizzate nell'ambiente on-premise. È necessario monitorare e ruotare il certificato X.509 quando si avvicina alla scadenza.

Risorse

Best practice correlate:

- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC02-BP03 Archiviazione e utilizzo dei segreti in modo sicuro

Documenti correlati:

- Nozioni di base su AWS Secrets Manager
- IAM Best Practices(Best Practice IAM)
- Provider di identità e federazione
- Soluzioni dei partner per la sicurezza: accesso e controllo degli accessi
- Credenziali di sicurezza temporanee
- Recupero dei report delle credenziali per l'Account AWS

Video correlati:

- Best practice per la gestione, il recupero e la rotazione di segreti su scala
- Managing user permissions at scale with AWS IAM Identity Center (Gestire le autorizzazioni degli utenti su larga scala con AWS SSO)
- Mastering identity at every layer of the cake

Esempi correlati:

- Well-Architected Lab - Automated IAM User Cleanup (Well-Architected Lab - Pulizia automatica degli utenti IAM)
- Well-Architected Lab - Automated Deployment of IAM Groups and Roles (Well-Architected Lab - Distribuzione automatica di gruppi e ruoli IAM)

SEC02-BP06 Impiego dei gruppi di utenti e degli attributi

Definire le autorizzazioni in base ai gruppi di utenti e agli attributi aiuta a ridurre il numero e la complessità delle policy, rendendo più semplice il raggiungimento del principio del privilegio minimo. Puoi usare i gruppi di utenti per gestire le autorizzazioni di molte persone in un'unica posizione, in

base alla funzione che svolgono nell'organizzazione. Gli attributi, come il reparto o la sede, possono fornire un ulteriore livello di portata dei permessi quando le persone svolgono una funzione simile ma per sottoinsiemi diversi di risorse.

Risultato desiderato: è possibile applicare le modifiche alle autorizzazioni in base alla funzione a tutti gli utenti che svolgono tale funzione. L'appartenenza al gruppo e gli attributi regolano le autorizzazioni degli utenti, riducendo la necessità di gestire le autorizzazioni a livello di singolo utente. I gruppi e gli attributi definiti nel gestore dell'identità digitale vengono propagati automaticamente agli ambienti AWS.

Anti-pattern comuni:

- Gestione delle autorizzazioni per singoli utenti e duplicazione tra più utenti.
- Definizione dei gruppi a un livello troppo alto, concessione di autorizzazioni troppo estese.
- Definizione di gruppi a un livello troppo granulare, che crea duplicazioni e confusione sull'appartenenza.
- Utilizzo di gruppi con autorizzazioni duplicate su sottoinsiemi di risorse quando è possibile utilizzare invece gli attributi.
- Nessuna gestione di gruppi, attributi e appartenenze attraverso un provider di identità standardizzato e integrato con gli ambienti AWS.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Le autorizzazioni AWS sono definite in documenti denominati policy associati a un principale, ad esempio un utente, un gruppo, un ruolo o una risorsa. Per la forza lavoro, ciò consente di definire i gruppi in base alla funzione svolta dagli utenti per l'organizzazione, anziché in base alle risorse a cui si accede. Ad esempio, un gruppo `WebAppDeveloper` può avere una policy allegata per la configurazione di un servizio, ad esempio Amazon CloudFront all'interno di un account di sviluppo. Un gruppo `AutomationDeveloper` può avere alcune autorizzazioni CloudFront in comune con il gruppo `WebAppDeveloper`. Queste autorizzazioni possono essere acquisite in una policy separata e associate a entrambi i gruppi, piuttosto che avere utenti di entrambe le funzioni che appartengono a un gruppo `CloudFront Access`.

Oltre ai gruppi, puoi utilizzare gli attributi per un ulteriore accesso all'ambito. Ad esempio, potresti avere un attributo `Project` per gli utenti del gruppo `WebAppDeveloper` per limitare l'accesso alle

risorse specifiche del loro progetto. L'uso di questa tecnica elimina la necessità di avere gruppi diversi per gli sviluppatori di applicazioni che lavorano su progetti diversi, se le loro autorizzazioni sono comunque le stesse. Il modo in cui si fa riferimento agli attributi nelle policy di autorizzazione si basa sulla loro origine, indipendentemente dal fatto che siano definiti come parte del protocollo di federazione (come SAML, OIDC o SCIM), come asserzioni SAML personalizzate o impostati all'interno di IAM Identity Center.

Passaggi dell'implementazione

1. Stabilisci dove definire gruppi e attributi.
   a. Seguendo le indicazioni contenute in SEC02-BP04 Fai affidamento su un provider di identità centralizzato, puoi determinare se definire gruppi e attributi all'interno del tuo provider di identità, in IAM Identity Center o utilizzando i gruppi IAM user in un account specifico.
2. Definisci i gruppi.
   a. Determina i tuoi gruppi in base alla funzione e all'ambito di accesso richiesti.
   b. Se definisci all'interno di IAM Identity Center, crea i gruppi e associa il livello di accesso desiderato utilizzando i set di autorizzazioni.
   c. Se definisci all'interno di un provider di identità esterno, determina se il provider supporta il protocollo SCIM e valuta la possibilità di abilitare il provisioning automatico all'interno di IAM Identity Center. Questa funzionalità sincronizza la creazione, l'appartenenza e l'eliminazione di gruppi tra il tuo provider e IAM Identity Center.
3. Definisci gli attributi.
   a. Se utilizzi un provider di identità esterno, entrambi i protocolli SCIM e SAML 2.0 forniscono determinati attributi per impostazione predefinita. È possibile definire e passare attributi aggiuntivi utilizzando le asserzioni SAML mediante il nome attributo `https://aws.amazon.com/SAML/Attributes/PrincipalTag`.
   b. Se definisci all'interno di IAM Identity Center, abilita la funzionalità di controllo degli accessi basato su attributi (ABAC) e definisci gli attributi come desiderato.
4. Autorizzazioni di ambito basate su gruppi e attributi.
   a. Considera la possibilità di includere nelle tue policy di autorizzazione condizioni che confrontino gli attributi del tuo principale con gli attributi delle risorse a cui si accede. Ad esempio, è possibile definire una condizione per consentire l'accesso a una risorsa solo se il valore di una chiave di condizione `PrincipalTag` corrisponde al valore di una chiave `ResourceTag` con lo stesso nome.

Risorse

Best practice correlate:

- SEC02-BP04 Fai affidamento su un provider di identità centralizzato
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- COST02-BP04 Implementazione di gruppi e ruoli

Documenti correlati:

- IAM Best Practices(Best Practice IAM)
- Manage Identities in IAM Identity Center
- What Is ABAC for AWS?
- ABAC In IAM Identity Center

Video correlati:

- Managing user permissions at scale with AWS IAM Identity Center
- Mastering identity at every layer of the cake

SEC 3. Come si gestisce l'autenticazione per persone e macchine?

Gestisci le autorizzazioni per controllare l'accesso alle identità di persone e macchine che richiedono l'accesso ad AWS e al tuo carico di lavoro. Le autorizzazioni controllano chi può accedere a cosa e a quali condizioni.

Best practice

- SEC03-BP01 Definizione dei requisiti di accesso
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- SEC03-BP03 Determinazione di un processo per l'accesso di emergenza
- SEC03-BP04 Riduzione delle autorizzazioni in modo continuo
- SEC03-BP05 Definizione dei guardrail per le autorizzazioni dell'organizzazione
- SEC03-BP06 Gestione degli accessi in base al ciclo di vita
- SEC03-BP07 Analisi dell'accesso pubblico e multi-account

- SEC03-BP08 Condivisione delle risorse in modo sicuro all'interno dell'organizzazione
- SEC03-BP09 Condivisione sicura delle risorse con terze parti

SEC03-BP01 Definizione dei requisiti di accesso

Ogni componente o risorsa del carico di lavoro deve essere accessibile da amministratori, utenti finali o altri componenti. Individua una definizione chiara di chi o cosa deve avere accesso a ciascun componente, quindi scegli il tipo di identità e il metodo di autenticazione e autorizzazione appropriati.

Anti-pattern comuni:

- Codifica fissa o archiviazione dei segreti nell'applicazione.
- Concessione di autorizzazioni personalizzate per ogni utente.
- Utilizzo di credenziali di lunga durata.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Ogni componente o risorsa del carico di lavoro deve essere accessibile da amministratori, utenti finali o altri componenti. Individua una definizione chiara di chi o cosa deve avere accesso a ciascun componente, quindi scegli il tipo di identità e il metodo di autenticazione e autorizzazione appropriati.

L'accesso regolare agli Account AWS dell'organizzazione viene fornito utilizzando l'accesso federato o un gestore dell'identità centralizzato. Occorre anche centralizzare la gestione delle identità e garantire la presenza di una procedura consolidata per integrare l'accesso ad AWS nel ciclo di vita dell'accesso dei dipendenti. Ad esempio, quando un dipendente passa a un ruolo lavorativo con un livello di accesso diverso, anche la sua appartenenza al gruppo deve cambiare per riflettere i nuovi requisiti di accesso.

Quando si definiscono i requisiti di accesso per le identità non umane, determina quali applicazioni e componenti devono accedere e come vengono concesse le autorizzazioni. L'utilizzo di ruoli IAM creati con il modello di accesso con privilegi minimi è un approccio consigliato. Le policy gestite da AWS forniscono le policy IAM predefinite che coprono la maggior parte dei casi d'uso comuni.

I servizi AWS, come AWS Secrets Manager e Archivio dei parametri AWS Systems Managerti consentono di scollegare i segreti dall'applicazione o dal carico di lavoro in modo sicuro nei casi in cui non è possibile utilizzare i ruoli IAM. In Secrets Manager puoi adottare la rotazione automatica

delle credenziali. Puoi usare Systems Manager per fare riferimento a parametri negli script, comandi, documenti SSM, configurazione e flussi di lavoro di automazione utilizzando il nome univoco specificato al momento della creazione del parametro.

Puoi usare AWS Identity and Access Management Roles Anywhere per ottenere credenziali di sicurezza temporanee in IAM per i carichi di lavoro eseguiti all'esterno di AWS. I tuoi carichi di lavoro possono utilizzare le stesse policy IAM e ruoli IAM che usi con le applicazioni AWS per accedere alle risorse AWS.

Ove possibile, prediligi le credenziali temporanee a breve termine rispetto a quelle statiche a lungo termine. Per gli scenari in cui gli utenti IAM devono avere l'accesso programmatico e credenziali a lungo termine, utilizza le ultime informazioni usate per la chiave di accesso per ruotare e rimuovere le chiavi di accesso.

Risorse

Documenti correlati:

- Il controllo dell'accesso basato su attributi (Attribute-Based Access Control, ABAC)
- AWS IAM Identity Center
- IAM Roles Anywhere
- AWS Managed policies for IAM Identity Center (Policy gestite da AWS per IAM Identity Center)
- AWS IAM policy conditions (Condizioni delle policy AWS IAM)
- Casi d'uso IAM
- Rimozione di credenziali non necessarie
- Lavorare con le policy
- How to control access to AWS resources based on Account AWS, OU, or organization (Come controllare l'accesso alle risorse AWS in base all'account, all'unità organizzativa o all'organizzazione AWS)
- Identify, arrange, and manage secrets easily using enhanced search in AWS Secrets Manager (Identificazione, organizzazione e gestione semplificate dei segreti con la ricerca avanzata di AWS Secrets Manager)

Video correlati:

- Become an IAM Policy Master in 60 Minutes or Less (Diventa un IAM Policy Master in 60 minuti)

- Separation of Duties, Least Privilege, Delegation, and CI/CD (Separazione dei compiti, privilegio minimo, delega e integrazione e distribuzione continua (CI/CD)
- Streamlining identity and access management for innovation (Semplificazione della gestione delle identità e degli accessi per l'innovazione)

SEC03-BP02 Concessione dell'accesso con privilegio minimo

È una best practice concedere alle identità soltanto il livello di accesso di cui hanno bisogno, specificando le operazioni che possono effettuare, le risorse su cui possono operare e a quali condizioni. Affidati ai gruppi e agli attributi di identità per impostare dinamicamente le autorizzazioni su vasta scala, anziché definire le autorizzazioni per i singoli utenti. Ad esempio, puoi concedere a un gruppo di sviluppatori le autorizzazioni per gestire solo le risorse del loro progetto. In questo modo, se uno sviluppatore lascia il progetto, il suo accesso viene automaticamente revocato senza modificare le policy di accesso sottostanti.

Risultato desiderato: gli utenti devono avere solo le autorizzazioni necessarie per portare a termine la loro attività. Gli utenti dovrebbero avere accesso solo agli ambienti di produzione per eseguire un'attività specifica in un intervallo temporale limitato e l'accesso dovrebbe essere revocato una volta completata l'attività. Le autorizzazioni devono essere revocate quando non sono più necessarie, incluso quando un utente passa a un progetto o a un ruolo professionale diversi. I privilegi di amministratore devono essere riservati a un piccolo gruppo di amministratori fidati. Le autorizzazioni devono essere riviste con regolarità per evitare che si accumulino. Account di sistemi o di macchine devono avere il numero minimo di autorizzazioni necessarie per portare a termine un'attività.

Anti-pattern comuni:

- L'impostazione predefinita è la concessione delle autorizzazioni di amministratore agli utenti.
- L'utilizzo dell'utente root per le attività quotidiane.
- Creazione di policy eccessivamente permissive, ma senza privilegi completi di amministratore.
- La mancata revisione delle autorizzazioni per capire se consentono l'accesso privilegio minimo.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

Secondo il principio del privilegio minimo le identità dovrebbero essere consentite solo per eseguire il numero minimo di azioni necessarie per completare un'attività specifica. In questo modo usabilità, efficienza e sicurezza sono bilanciate. Seguendo questo principio si limitano gli accessi indesiderati

e si può monitorare chi accede a quali risorse. Gli utenti e i ruoli IAM non hanno autorizzazioni per impostazione predefinita. L'utente root ha accesso completo per impostazione predefinita e dovrebbe essere controllato e monitorato con zelo, nonché usato solo per le attività che richiedono l'accesso root.

Le policy IAM sono utilizzate in modo esplicito per concedere le autorizzazioni ai ruoli IAM o a risorse specifiche. Ad esempio, le policy basate su identità possono essere collegate ai gruppi IAM, mentre i bucket S3 possono essere controllati da policy basate su risorse.

Quando crei e colleghi una policy IAM, puoi specificare le azioni del servizio, le risorse e le condizioni che devono essere vere affinché AWS consenta o neghi l'accesso. AWS supporta una varietà di condizioni che contribuiscono a ridurre l'accesso. Ad esempio, se usi la chiave di condizione `PrincipalOrgID`, puoi non autorizzare le operazioni se il richiedente non è parte della tua AWS Organization.

Puoi anche controllare le richieste effettuate dai servizi AWS per tuo conto, ad esempio AWS CloudFormation per la creazione di una funzione AWS Lambda, utilizzando la chiave di condizione `CalledVia`. Dovresti avere tipi diversi di policy su più livelli per definire un livello di difesa ben radicato e limitare le autorizzazioni complessive dei tuoi utenti. Puoi anche limitare le autorizzazioni che possono essere concesse e a quali condizioni. Ad esempio, puoi consentire ai team delle applicazioni di creare le proprie policy IAM per i sistemi che creano, ma devi anche applicare un limite delle autorizzazioni per impostare un limite massimo di autorizzazioni che il sistema può ricevere.

Passaggi dell'implementazione

- Implementazione di policy con privilegi minimi: assegna policy di accesso con privilegi minimi a gruppi e ruoli IAM in modo da rispecchiare il ruolo o la funzione dell'utente che hai definito.
  - Policy di base sull'uso delle API: un modo per stabilire le autorizzazioni necessarie consiste nell'analisi dei log AWS CloudTrail. Questa revisione consente di creare autorizzazioni personalizzate in base alle azioni che l'utente deve realmente eseguire in AWS. IAM Access Analyzer può generare automaticamente una policy IAM basata su attività. Puoi usare IAM Access Advisor a livello di account o di organizzazione per monitorare le ultime informazioni consultate per una policy specifica.
- Prendi in considerazione l'uso di policy gestite da AWS per le funzioni dell'attività. Quando inizi a creare policy di autorizzazioni dettagliate, può essere difficile sapere da dove iniziare. AWS ha policy gestite per ruoli professionali comuni, ad esempio contabili, amministratori di database e data scientist. Queste policy possono contribuire a limitare l'accesso degli utenti e, al contempo, definiscono come implementare le policy di privilegio minimo.

- Rimuovi le autorizzazioni superflue: rimuovi le autorizzazioni non necessarie e rivedi quelle eccessivamente permissive. La generazione di policy di IAM Access Analyzer può essere utile per perfezionare le policy relative alle autorizzazioni.
- Verifica che gli utenti abbiano un accesso limitato agli ambienti di produzione: gli utenti possono accedere agli ambienti di produzione solo se hanno un caso d'uso valido. Una volta eseguite le attività specifiche che richiedono l'accesso alla produzione, l'accesso dell'utente deve essere revocato. Limitare l'accesso agli ambienti di produzione contribuisce a evitare eventi indesiderati con impatto sulla produzione e contiene gli effetti di accessi involontari.
- Considerazioni sui limiti delle autorizzazioni: un limite delle autorizzazioni è una caratteristica avanzata per utilizzare una policy gestita che imposta il numero massimo di autorizzazioni che una policy basata su identità può concedere a un'entità IAM. Il limite delle autorizzazioni di un'entità le permette di eseguire solo le operazioni consentite dalle policy basate su identità e dai limiti delle autorizzazioni.
- Prendi in considerazione i tag delle risorse per le autorizzazioni: un modello di controllo degli accessi basato su attributi che usa tag delle risorse ti consente di concedere l'accesso in base a scopo delle risorse, proprietario, ambiente e altri criteri. Ad esempio, puoi usare tag di risorse per diversificare gli ambienti di produzione e sviluppo. Tramite questi tag puoi limitare gli sviluppatori all'ambiente di sviluppo. Abbinando policy su tag e autorizzazioni, puoi ottenere l'accesso a risorse dettagliate senza dover definire policy personalizzate e complesse per ogni funzione professionale.
- Usa policy di controllo dei servizi per AWS Organizations. Le policy di controllo dei servizi monitorano centralmente il numero massimo di autorizzazioni disponibili per gli account membri della tua organizzazione. È importante notare che le policy di controllo dei servizi consentono di limitare le autorizzazioni dell'utente root negli account membri. Considera anche la possibilità di usare AWS Control Tower, che offre controlli gestiti prescrittivi che arricchiscono AWS Organizations. Puoi anche definire i tuoi controlli in Control Tower.
- Stabilisci una policy del ciclo di vita dell'utente per la tua organizzazione: le policy del ciclo di vita dell'utente definiscono attività da eseguire quando gli utenti eseguono l'onboarding su AWS, cambiano ruolo o ambito professionale o non hanno più bisogno di accedere a AWS. Le revisioni delle autorizzazioni devono essere eseguite in ogni fase del ciclo di vita di un utente per verificare che siano sufficientemente restrittive e per evitare che si accumulino.
- Stabilisci un piano per analizzare le autorizzazioni con regolarità ed eventualmente rimuovere quelle non necessarie: dovresti periodicamente analizzare l'accesso degli utenti per verificare che non abbiano autorizzazioni troppo permissive. AWS Config e IAM Access Analyzer può essere utile in fase di audit delle autorizzazioni utente.

- Definisci una matrice dei ruoli professionali: una matrice dei ruoli professionali mostra i diversi ruoli e livelli di accesso richiesti all'interno della tua presenza in AWS. Tramite una matrice dei ruoli professionali puoi definire e separare le autorizzazioni in base alle responsabilità degli utenti all'interno dell'organizzazione. Usa i gruppi invece di applicare le autorizzazioni direttamente ai singoli utenti o ruoli.

Risorse

Documenti correlati:

- Assegnare il privilegio minimo
- Limiti delle autorizzazioni per le entità IAM
- Tecniche per la scrittura di policy IAM con privilegio minimo
- IAM Access Analyzer semplifica l'implementazione delle autorizzazioni con privilegio minimo generando IAM policy basate sull'attività di accesso
- Delegare la gestione delle autorizzazioni agli sviluppatori tramite i limiti delle autorizzazioni IAM
- Perfezionamento delle autorizzazioni in AWS utilizzando le informazioni sull'ultimo accesso
- Tipi di policy IAM e quando utilizzarle
- Test delle policy IAM con il simulatore di policy IAM
- Guardrail in AWS Control Tower
- Architetture Zero Trust: una prospettiva AWS
- Come implementare il principio del privilegio minimo con CloudFormation StackSets
- Il controllo dell'accesso basato su attributi (Attribute-Based Access Control, ABAC)
- Riduzione dell'ambito di applicazione della policy mediante visualizzazione dell'attività dell'utente
- Visualizzazione dell'accesso al ruolo
- Utilizza l'applicazione di tag per organizzare il tuo ambiente e per promuovere la responsabilità
- Strategie di applicazione di tag AWS
- Applicazione di tag alle risorse AWS

Video correlati:

- Next-generation permissions management (Gestione delle autorizzazioni di ultima generazione)

- Zero Trust: una prospettiva AWS
- Come posso utilizzare i limiti delle autorizzazioni per limitare utenti e ruoli e impedire l'escalation dei privilegi?

Esempi correlati:

- Laboratorio: limiti delle autorizzazioni IAM per delegare la creazione di ruoli
- Laboratorio: controllo degli accessi basato su tag IAM per EC2

SEC03-BP03 Determinazione di un processo per l'accesso di emergenza

Crea un processo che consenta l'accesso di emergenza ai tuoi carichi di lavoro nell'improbabile eventualità che si verifichi un problema con il tuo provider di identità centralizzato.

È necessario progettare processi per diverse modalità di guasto che possono causare un evento di emergenza. Ad esempio, in circostanze normali, gli utenti della tua forza lavoro si federano nel cloud utilizzando un provider di identità centralizzato (SEC02-BP04) per gestire i propri carichi di lavoro. Tuttavia, se il tuo provider di identità centralizzato genera un errore o la configurazione per la federazione nel cloud viene modificata, gli utenti della tua forza lavoro potrebbero non essere in grado di federarsi nel cloud. Un processo di accesso di emergenza consente agli amministratori autorizzati di accedere alle risorse cloud tramite mezzi alternativi (come una forma alternativa di federazione o l'accesso diretto degli utenti) per risolvere problemi relativi alla configurazione della federazione o ai carichi di lavoro. Il processo di accesso di emergenza viene utilizzato fino al ripristino del normale meccanismo di federazione.

Risultato desiderato:

- Hai definito e documentato le modalità di guasto che costituiscono un'emergenza: considera le circostanze normali e i sistemi da cui dipendono gli utenti per gestire i loro carichi di lavoro. Considera quali guasti possono interessare ognuna di queste dipendenze e causare una situazione di emergenza. Puoi trovare le domande e le best practice nel Principio di base dell'affidabilità , utile per identificare le modalità di errore e progettare sistemi più resilienti per ridurre al minimo la probabilità di guasti.
- Hai documentato i passaggi da seguire per confermare che un guasto costituisce un'emergenza. Ad esempio, puoi richiedere agli amministratori di identità di controllare lo stato dei provider di identità primari e di standby e, se entrambi non sono disponibili, dichiarare un evento di emergenza per guasto del provider di identità.

- È stato definito un processo di accesso di emergenza specifico per ogni tipo di modalità di emergenza o di guasto. Essere specifici può ridurre la tentazione da parte degli utenti di abusare di un processo generale per tutti i tipi di emergenze. I processi di accesso di emergenza descrivono le circostanze in cui ogni processo deve essere o non deve essere utilizzato e indicano processi alternativi che possono essere applicati.
- I tuoi processi sono ben documentati con istruzioni e playbook dettagliati che possono essere seguiti in modo rapido ed efficiente. Ricorda che un evento di emergenza può essere un momento stressante per i tuoi utenti, che potrebbero essere sotto pressione per motivi di tempo, quindi progetta il tuo processo in modo che sia il più semplice possibile.

Anti-pattern comuni:

- Non si dispone di procedure di accesso di emergenza ben documentate e collaudate. Gli utenti non sono preparati per un'emergenza e seguono processi improvvisati quando si verifica un evento di emergenza.
- I processi di accesso di emergenza dipendono dagli stessi sistemi (come un provider di identità centralizzato) dei normali meccanismi di accesso. Ciò significa che il guasto di un sistema di questo tipo può influire sui normali meccanismi di accesso e di emergenza e compromettere la capacità di ripristino dall'errore.
- I processi di accesso di emergenza vengono utilizzati in situazioni non di emergenza. Ad esempio, gli utenti utilizzano spesso in modo improprio i processi di accesso di emergenza poiché trovano più facile apportare modifiche direttamente piuttosto che inviarle tramite una pipeline.
- I processi di accesso di emergenza non generano log sufficienti per controllare i processi oppure i log non vengono monitorati per segnalare un potenziale uso improprio dei processi.

Vantaggi dell'adozione di questa best practice:

- Grazie a processi di accesso di emergenza ben documentati e collaudati, puoi ridurre il tempo impiegato dagli utenti per rispondere a un evento di emergenza e risolverlo. Ciò può comportare una riduzione dei tempi di inattività e una maggiore disponibilità dei servizi forniti ai clienti.
- È possibile tenere traccia di ogni richiesta di accesso di emergenza e rilevare e avvisare in caso di tentativi non autorizzati di uso improprio del processo per eventi non di emergenza.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Questa sezione fornisce indicazioni per la creazione di processi di accesso di emergenza per diverse modalità di errore relative ai carichi di lavoro distribuiti su AWS, a partire da linee guida comuni che si applicano a tutte le modalità di errore fino a linee guida specifiche in base al tipo di errore.

Linee guida comuni per tutte le modalità di errore

Nella progettazione di un processo di accesso di emergenza per una modalità di errore, tieni presente quanto segue:

- Documenta i prerequisiti e i presupposti del processo: quando il processo deve e non deve essere utilizzato. Aiuta a descrivere in dettaglio la modalità di errore e a documentare le ipotesi, come lo stato di altri sistemi correlati. Ad esempio, il processo per la modalità di errore 2 presuppone che il provider di identità sia disponibile, ma la configurazione in AWS è stata modificata o è scaduta.
- Crea preliminarmente le risorse necessarie per il processo di accesso di emergenza (SEC10-BP05). Ad esempio, crea preliminarmente l'accesso di emergenza a un Account AWS con ruoli e IAM users e in tutti gli account del carico di lavoro creando ruoli IAM multi-account. Ciò assicura che queste risorse siano pronte e disponibili quando si verifica un evento di emergenza. Creando preliminarmente le risorse, non si ha alcuna dipendenza dalle API del piano di controllo di AWS (utilizzate per creare e modificare risorse AWS), che potrebbero non essere disponibili in caso di emergenza. Inoltre, creando preliminarmente le risorse IAM, non è necessario tenere conto di potenziali ritardi dovuti alla coerenza finale.
- Includi i processi di accesso di emergenza nei tuoi piani di gestione degli incidenti (SEC10-BP02). Documenta in che modo viene tenuta traccia degli eventi di emergenza e come essi vengono comunicati ad altri membri dell'organizzazione, come i team di pari livello, la leadership e, se applicabile, esternamente ai clienti e ai partner aziendali.
- Definisci il processo di richiesta di accesso di emergenza nel tuo sistema di flusso di lavoro esistente, se ne hai uno, per le richieste di assistenza. In genere, tali sistemi di flusso di lavoro consentono di creare moduli di acquisizione per raccogliere informazioni sulla richiesta, tenere traccia della richiesta in ogni fase del flusso di lavoro e aggiungere passaggi di approvazione automatici e manuali. Collega ogni richiesta a un evento di emergenza corrispondente tracciato nel tuo sistema di gestione degli incidenti. Disporre di un sistema uniforme per gli accessi di emergenza consente di tenere traccia di tali richieste in un unico sistema, analizzare le tendenze di utilizzo e migliorare i processi.
- Verifica che i processi di accesso di emergenza possano essere avviati solo da utenti autorizzati e richiedano l'approvazione dei colleghi o dei manager dell'utente, a seconda dei casi. Il processo

di approvazione deve funzionare efficacemente sia all'interno che al di fuori dell'orario lavorativo. Definisci in che modo le richieste di approvazione possono essere eseguite da approvatori secondari, qualora gli approvatori principali non fossero disponibili, e come vengono inoltrate lungo la catena di gestione fino all'approvazione.

- Verifica che il processo generi log di controllo ed eventi dettagliati per i tentativi riusciti e falliti di ottenere l'accesso di emergenza. Monitora sia il processo di richiesta sia il meccanismo di accesso di emergenza per rilevare usi impropri o accessi non autorizzati. Metti in correlazione l'attività con gli eventi di emergenza in corso dal tuo sistema di gestione degli incidenti e avvisa quando le azioni si verificano al di fuori dei periodi di tempo previsti. Ad esempio, devi monitorare e inviare avvisi in merito ad attività nell'Account AWS di accesso di emergenza, poiché non dovrebbe mai essere utilizzato per le normali operazioni.
- Testa periodicamente i processi di accesso di emergenza per verificare che i passaggi siano chiari e garantire il livello di accesso corretto in modo rapido ed efficiente. I processi di accesso di emergenza devono essere testati nell'ambito delle simulazioni di risposta agli incidenti (SEC10-BP07) e test di ripristino di emergenza (REL13-BP03).

Modalità di errore 1: il provider di identità utilizzato per la federazione dell'accesso ad AWS non è disponibile

Come descritto in SEC02-BP04 Fai affidamento su un provider di identità centralizzato, ti consigliamo di affidarti a un provider di identità centralizzato per federare gli utenti della tua forza lavoro e garantire loro l'accesso agli Account AWS. È possibile federare l'accesso agli Account AWS a più utenti AWS all'interno dell'organizzazione utilizzando IAM Identity Center, oppure federare l'accesso individuale agli Account AWS utilizzando IAM. In entrambi i casi, gli utenti della forza lavoro si autenticano con il provider di identità centralizzato prima di essere reindirizzati a un endpoint di accesso AWS per l'autenticazione unica.

Nell'improbabile eventualità che il provider di identità centralizzato non sia disponibile, gli utenti della tua forza lavoro non possono federarsi per accedere agli Account AWS o gestire i propri carichi di lavoro. In questo caso critico, puoi fornire un processo di accesso di emergenza a cui un piccolo gruppo di amministratori può accedere agli Account AWS per eseguire attività urgenti per le quali non è possibile attendere che i tuoi provider di identità centralizzati tornino online. Ad esempio, il tuo provider di identità non è disponibile per 4 ore e durante quel periodo devi modificare i limiti massimi di un gruppo Amazon EC2 Auto Scaling in un account di produzione per gestire un picco imprevisto nel traffico dei clienti. Gli amministratori di emergenza devono seguire la procedura di accesso di emergenza per accedere a un Account AWS di produzione specifico e apportare le modifiche necessarie.

Il processo di accesso di emergenza si basa su un accesso di emergenza a un Account AWS creato preliminarmente, che viene utilizzato esclusivamente per questo tipo di accessi e dispone di risorse AWS (come ruoli IAM e IAM users) per supportare il processo di accesso di emergenza. Durante le normali operazioni, nessuno deve accedere all'account di accesso di emergenza ed è necessario monitorare e fornire avvisi riguardo a usi impropri di questo account (per maggiori dettagli, vedi la sezione precedente Linee guida comuni).

L'account di accesso di emergenza dispone di ruoli di accesso di emergenza IAM con autorizzazioni per assumere ruoli multi-account negli Account AWS che richiedono l'accesso di emergenza. Questi ruoli IAM sono creati preliminarmente e configurati con policy di attendibilità che valutano i ruoli IAM dell'account di emergenza come attendibili.

Per il processo di accesso di emergenza è possibile utilizzare uno dei seguenti approcci:

- Creare preliminarmente un set di IAM users per gli amministratori di emergenza nell'account di accesso di emergenza con password complesse e token MFA associati. Questi IAM users dispongono delle autorizzazioni per assumere i ruoli IAM che consentono l'accesso multi-account all'Account AWS per cui è richiesto l'accesso di emergenza. Ti consigliamo di creare il minor numero possibile di utenti di questo tipo e di assegnare ogni utente a un unico amministratore di emergenza. Durante un'emergenza, un utente amministratore di emergenza accede all'account di accesso di emergenza utilizzando la propria password e il codice token MFA, passa al ruolo IAM di accesso di emergenza nell'account di emergenza e infine passa al ruolo IAM di accesso di emergenza nell'account del carico di lavoro per eseguire l'azione di modifica di emergenza. Il vantaggio di questo approccio è che ogni IAM user è assegnato a un amministratore di emergenza e puoi sapere quale utente ha effettuato l'accesso esaminando gli eventi CloudTrail. Lo svantaggio è che è necessario mantenere più IAM users con le relative password di lunga durata e i token MFA associati.
- È possibile utilizzare l'accesso di emergenza come utente root dell'Account AWS per accedere all'account di emergenza, assumere il ruolo IAM per l'accesso di emergenza e poi il ruolo multi-account nell'account del carico di lavoro. È consigliabile impostare una password sicura e più token MFA per l'utente root. Consigliamo inoltre di archiviare la password e i token MFA in un archivio di credenziali aziendali sicuro, che applichi policy di autenticazione e autorizzazione avanzate. Proteggi i fattori di reimpostazione della password e del token MFA: imposta l'indirizzo e-mail dell'account su una lista di distribuzione e-mail monitorata dagli amministratori della sicurezza del cloud e il numero di telefono dell'account su un numero di telefono condiviso anch'esso monitorato dagli amministratori della sicurezza. Il vantaggio di questo approccio è che esiste un solo set di credenziali utente root da gestire. Lo svantaggio è che, trattandosi di un utente condiviso, più

amministratori hanno la possibilità di accedere come utente root. Controlla il log eventi della tua vault aziendale per identificare quale amministratore ha utilizzato la password dell'utente root.

Modalità di errore 2: la configurazione del provider di identità su AWS è stata modificata o è scaduta

Per consentire agli utenti della tua forza lavoro di effettuare l'accesso federato agli Account AWS, puoi configurare il IAM Identity Center con un provider di identità esterno o creare un provider di identità IAM (SEC02-BP04). In genere, la configurazione viene effettuata importando un documento XML di metadati SAML fornito dal provider di identità. Il documento XML di metadati include un certificato X.509 corrispondente a una chiave privata utilizzata dal provider di identità per firmare le sue asserzioni SAML.

Queste configurazioni lato AWS possono essere modificate o eliminate per errore da un amministratore. In un altro scenario, può accadere che il certificato X.509 importato in AWS sia scaduto e che un nuovo XML di metadati con un nuovo certificato non sia ancora stato importato in AWS. In entrambi gli scenari, la federazione degli utenti della forza lavoro per accedere ad AWS può essere interrotta, costituendo una situazione di emergenza.

In un caso di emergenza di questo tipo, puoi fornire agli amministratori di identità l'accesso ad AWS per risolvere i problemi di federazione. Ad esempio, l'amministratore delle identità utilizza la procedura di accesso di emergenza per accedere a un Account AWS, passa a un ruolo nell'account amministratore del Centro di identità e riattiva la federazione aggiornando la configurazione del provider di identità esterno e importando l'ultimo documento XML di metadati SAML rilasciato dal provider di identità. Una volta ristabilita la federazione, gli utenti della forza lavoro continuano a utilizzare il normale processo operativo per federare l'accesso ai propri account di carico di lavoro.

È possibile seguire gli approcci descritti nella sezione precedente Modalità di errore 1 per creare un processo di accesso di emergenza. Puoi concedere le autorizzazioni con il privilegio minimo agli amministratori di identità per accedere solo all'account amministratore di Centro di identità ed eseguire azioni sul Centro di identità in quell'account.

Modalità di errore 3: blocco del Centro di identità

Nell'improbabile eventualità di un blocco di IAM Identity Center o di una Regione AWS, ti consigliamo di eseguire una configurazione per fornire l'accesso temporaneo alla AWS Management Console.

Il processo di accesso di emergenza utilizza la federazione diretta rilasciata dal provider di identità a un ruolo IAM per accedere a un account di emergenza. Per informazioni dettagliate sulle

considerazioni relative al processo e alla progettazione, consulta Configurare l'accesso di emergenza alla AWS Management Console.

Passaggi dell'implementazione

Passaggi comuni per tutte le modalità di errore

- Crea un Account AWS dedicato per gli accessi di emergenza. Crea preliminarmente le risorse IAM necessarie nell'account, come i ruoli IAM o gli utenti IAM users, e, in modo facoltativo, i provider di identità IAM. Inoltre, crea preliminarmente ruoli IAM mutli-account negli Account AWS del carico di lavoro dotati di relazioni di fiducia con i ruoli IAM corrispondenti nell'account di accesso di emergenza. Puoi utilizzare AWS CloudFormation StackSets con AWS Organizations per creare tali risorse negli account dei membri della tua organizzazione.

- Crea Policy di controllo dei servizi AWS Organizations (SCP) per negare l'eliminazione e la modifica dei ruoli IAM multi-account negli Account AWS dei membri.
- Abilita CloudTrail per l'accesso di emergenza a un Account AWS e invia gli eventi di trail a un bucket S3 centrale nella raccolta di log relativa all'Account AWS. Se utilizzi AWS Control Tower per configurare e gestire il tuo ambiente AWS multi-account, ogni account che crei utilizzando AWS Control Tower o a cui ti iscrivi in AWS Control Tower ha CloudTrail abilitato per impostazione predefinita e viene inviato a un bucket S3 in un Account AWS con archivio di log dedicato.
- Monitora l'attività dell'account di accesso di emergenza creando regole EventBridge coerenti con l'accesso alla console e all'attività dell'API da parte dei ruoli IAM di emergenza. Invia notifiche al tuo centro operativo di sicurezza quando si verificano attività al di fuori di un evento di emergenza in corso e di cui hai traccia nel tuo sistema di gestione degli incidenti.

Passaggi aggiuntivi per la Modalità di errore 1: il provider di identità utilizzato per la federazione dell'accesso ad AWS non è disponibile; per la Modalità di errore 2: la configurazione del provider di identità su AWS è stata modificata o è scaduta

- Crea preliminarmente le risorse in base al meccanismo scelto per l'accesso di emergenza:
  - Utilizza IAM users: crea preliminarmente IAM users con password complesse e dispositivi MFA associati.
  - Usa l'utente root dell'account di emergenza: configura l'utente root con una password sicura e archivia la password nel tuo archivio di credenziali aziendali. Associa più dispositivi MFA fisici all'utente root e archivia i dispositivi in posizioni a cui i membri del team di amministrazione delle emergenze possono accedere rapidamente.

Passaggi aggiuntivi per la Modalità di errore 3: blocco del Centro di identità

- Come spiegato nei dettagli in Configurare l'accesso di emergenza alla AWS Management Console, per l'accesso di emergenza a un Account AWS, crea un provider di identità IAM per abilitare la federazione SAML diretta dal tuo provider di identità.
- Crea gruppi operativi di emergenza nel tuo IdP senza membri.
- Crea ruoli IAM corrispondenti ai gruppi operativi di emergenza nell'account di accesso di emergenza.

Risorse

Best practice Well-Architected correlate:

- SEC02-BP04 Fai affidamento su un provider di identità centralizzato
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- SEC10-BP02 Sviluppo di piani di gestione degli incidenti
- SEC10-BP07 Esecuzione di giornate di gioco

Documenti correlati:

- Set up emergency access to the AWS Management Console
- Abilitazione degli utenti federati SAML 2.0 per accedere a AWS Management Console
- Break glass access

Video correlati:

- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center
- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive

Esempi correlati:

- AWS Break Glass Role
- AWS customer playbook framework
- AWS incident response playbook samples

SEC03-BP04 Riduzione delle autorizzazioni in modo continuo

Man mano che i team determinano gli accessi necessari, rimuovi i permessi non necessari e stabilisci processi di revisione per ottenere i permessi del privilegio minimo. Monitora costantemente e rimuovi le identità e le autorizzazioni inutilizzate per l'accesso sia umano che automatico.

Risultato desiderato: le policy di autorizzazione devono attenersi al principio del privilegio minimo. Man mano che le mansioni e i ruoli vengono definiti meglio, è necessario rivedere le policy di autorizzazione per eliminare le autorizzazioni non necessarie. Questo approccio riduce la portata dell'impatto nel caso in cui le credenziali vengano inavvertitamente esposte o si acceda in altro modo senza autorizzazione.

Anti-pattern comuni:

- L'impostazione predefinita è la concessione delle autorizzazioni di amministratore agli utenti.
- Creazione di policy eccessivamente permissive, ma senza privilegi completi di amministratore.
- Mantenimento delle policy di autorizzazione anche quando non sono più necessarie.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Quando i team e i progetti sono in fase iniziale, le policy di autorizzazione permissiva possono essere utilizzate per stimolare l'innovazione e l'agilità. Ad esempio, in un ambiente di sviluppo o di test, gli sviluppatori possono avere accesso a un'ampia gamma di servizi AWS. Si consiglia di valutare costantemente gli accessi e di limitare l'accesso solo ai servizi e alle azioni di servizio necessari per completare il lavoro in corso. Raccomandiamo questa valutazione sia per l'identità umana che per quella macchina. Le identità macchina, talvolta chiamate account di sistema o di servizio, sono identità che consentono ad AWS di accedere ad applicazioni o server. Questo accesso è particolarmente importante in un ambiente di produzione, dove autorizzazioni troppo permissive possono avere un ampio impatto e potenzialmente esporre i dati dei clienti.

AWS offre diversi metodi per identificare utenti, ruoli, autorizzazioni e credenziali non utilizzati. AWS può anche aiutare ad analizzare l'attività di accesso degli utenti e dei ruoli IAM, comprese le chiavi di accesso associate, e l'accesso alle risorse AWS, come gli oggetti nei bucket Amazon S3. La generazione di policy di AWS Identity and Access Management Access Analyzer può aiutare a creare policy di autorizzazione restrittive in base ai servizi e alle azioni effettive con cui interagisce un principale. Controllo dell'accesso basato sugli attributi (ABAC) può aiutare a semplificare la gestione

delle autorizzazioni, in quanto è possibile fornire autorizzazioni agli utenti utilizzando i loro attributi invece di allegare le policy di autorizzazione direttamente a ciascun utente.

Passaggi dell'implementazione

- Utilizza AWS Identity and Access Management Access Analyzer: IAM Access Analyzer aiuta a identificare le risorse nell'organizzazione e negli account, come ad esempio bucket Amazon Simple Storage Service (Amazon S3) o ruoli IAM che sono condivisi con un'entità esterna.
- Utilizza la generazione della policy IAM Access Analyzer: la generazione della policy IAM Access Analyzer aiuta a creare policy di autorizzazione granulari basate su un utente IAM o su un'attività di accesso del ruolo.
- Stabilisci una tempistica accettabile e una policy di utilizzo per gli utenti e i ruoli IAM: utilizza il timestamp dell'ultimo accesso per identificare gli utenti e i ruoli inutilizzati e rimuoverli. Rivedi le informazioni sull'ultimo accesso al servizio e sull'ultima azione per identificare e delimitare le autorizzazioni per specifici utenti e ruoli. Ad esempio, puoi utilizzare le informazioni sull'ultimo accesso per identificare le azioni specifiche di Amazon S3 richieste dal ruolo dell'applicazione e delimitare l'accesso del ruolo solo a tali azioni. Le funzionalità relative alle informazioni sull'ultimo accesso sono disponibili nella AWS Management Console e consentono di incorporarle in modo programmatico nei flussi di lavoro dell'infrastruttura e negli strumenti automatizzati.
- Considera la registrazione degli eventi di dati in AWS CloudTrail: per impostazione predefinita, CloudTrail non registra eventi di dati come le attività a livello di oggetto Amazon S3 (ad esempio, `GetObject` e `DeleteObject`) o le attività della tabella Amazon DynamoDB (ad esempio, `PutItem` e `DeleteItem`). Considera la possibilità di abilitare la registrazione di questi eventi per stabilire quali utenti e ruoli devono accedere a specifici oggetti Amazon S3 o elementi di tabelle DynamoDB.

Risorse

Documenti correlati:

- Assegnare il privilegio minimo
- Rimozione di credenziali non necessarie
- Cosa è AWS CloudTrail?
- Working with Policies (Gestire le policy)
- Registrazione e monitoraggio in DynamoDB
- Abilitazione della registrazione di eventi CloudTrail per bucket e oggetti Amazon S3

- Recupero dei report delle credenziali per l'Account AWS

Video correlati:

- Become an IAM Policy Master in 60 Minutes or Less (Diventa un IAM Policy Master in 60 minuti)
- Separation of Duties, Least Privilege, Delegation, and CI/CD (Separazione dei compiti, privilegio minimo, delega e integrazione e distribuzione continua (CI/CD)
- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive [AWS re:Inforce 2022 - Approfondimento su AWS Identity and Access Management (IAM)]

SEC03-BP05 Definizione dei guardrail per le autorizzazioni dell'organizzazione

Utilizza i guardrail delle autorizzazioni per ridurre l'ambito delle autorizzazioni disponibili che possono essere concesse ai principali. La catena di valutazione delle policy di autorizzazione include i guardrail per determinare le autorizzazioni effettive di un principale quando prende decisioni di autorizzazione. È possibile definire i guardrail utilizzando un approccio basato sui livelli. Applica alcuni guardrail in modo esteso all'intera organizzazione e applicane altri in modo granulare alle sessioni di accesso temporaneo.

Risultato desiderato: si ha un chiaro isolamento degli ambienti utilizzando Account AWS separati. Le policy di controllo dei servizi (SCP) vengono utilizzate per definire i guardrail delle autorizzazioni a livello di organizzazione. I guardrail più estesi sono impostati ai livelli gerarchici più vicini alla radice dell'organizzazione, mentre i guardrail più rigidi sono impostati più vicino al livello dei singoli account. Se supportate, le policy sulle risorse definiscono le condizioni che un principale deve soddisfare per ottenere l'accesso a una risorsa. Le policy per le risorse, inoltre, definiscono l'insieme delle azioni consentite, ove appropriato. I limiti delle autorizzazioni sono posti sui principali che gestiscono le autorizzazioni del carico di lavoro, delegando la gestione delle autorizzazioni ai singoli proprietari del carico di lavoro.

Anti-pattern comuni:

- Creare un membro Account AWS all'interno di una AWS Organization, ma non utilizzare gli SCP per limitare l'uso e le autorizzazioni disponibili per le loro credenziali root.
- Assegnare i permessi in base al privilegio minimo, senza però porre guardrail sull'insieme massimo di permessi che possono essere concessi.
- Affidarsi alla base del rifiuto implicito di AWS IAM per limitare le autorizzazioni, confidando nel fatto che le policy non concedano un'autorizzazione esplicita indesiderata.

- Eseguire più ambienti di carico di lavoro nello stesso Account AWS e affidarsi quindi a meccanismi come VPC, tag o policy sulle risorse per applicare i limiti delle autorizzazioni.

Vantaggi della definizione di questa best practice: i guardrail delle autorizzazioni contribuiscono a creare la certezza che non possano essere concesse autorizzazioni indesiderate, anche quando una policy di autorizzazione tenta di farlo. Ciò può semplificare la definizione e la gestione delle autorizzazioni riducendo l'ambito massimo delle autorizzazioni da prendere in considerazione.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Ti consigliamo di utilizzare un approccio basato sui livelli per definire i guardrail di autorizzazione per la tua organizzazione. Questo approccio riduce sistematicamente il set massimo di autorizzazioni possibili man mano che vengono applicati livelli aggiuntivi. Ciò consente di concedere l'accesso in base al principio del privilegio minimo, riducendo il rischio di accessi non intenzionali dovuti a un'errata configurazione delle policy.

Il primo passo per definire i guardrail dei permessi è isolare i carichi di lavoro e gli ambienti in Account AWS separati. I principali di un account non possono accedere alle risorse di un altro account senza l'autorizzazione esplicita in tal senso, anche quando entrambi gli account si trovano nella stessa organizzazione AWS o nella stessa unità organizzativa (UO). Puoi utilizzare le unità organizzative per raggruppare gli account che desideri amministrare come una singola unità.

Il passaggio successivo consiste nel ridurre il set massimo di autorizzazioni che è possibile concedere ai principali all'interno degli account dei membri dell'organizzazione. A tale scopo, puoi utilizzare le policy di controllo dei servizi, che puoi applicare a un'unità organizzativa o a un account. Le SCP possono applicare controlli di accesso comuni, ad esempio limitare l'accesso a Regioni AWS specifiche, aiutare a prevenire l'eliminazione di risorse o disabilitare azioni di servizio potenzialmente rischiose. Le SCP applicate alla radice dell'organizzazione influiscono solo sugli account dei membri, non sull'account di gestione. Le SCP regolano solo i principali all'interno della tua organizzazione. Le tue SCP non regolano i principali esterni alla tua organizzazione che accedono alle tue risorse.

Un ulteriore passaggio consiste nell'utilizzare le policy sulla risorsa IAM per definire le azioni disponibili che è possibile intraprendere sulle risorse che governano, insieme a tutte le condizioni che il principale ad interim deve soddisfare. Ciò può comprendere tutte le azioni a condizione che il responsabile faccia parte dell'organizzazione (utilizzando la chiave di condizione PrincipalOrgid)

oppure può essere granulare, consentendo solo azioni specifiche da parte di un ruolo IAM specifico. Puoi adottare un approccio simile con le condizioni nelle policy di attendibilità del ruolo IAM. Se una policy di attendibilità di una risorsa o di un ruolo nomina esplicitamente un principale nello stesso account del ruolo o della risorsa che governa, tale principale non ha bisogno di una policy IAM associata che conceda le stesse autorizzazioni. Se il principale si trova in un account diverso dalla risorsa, deve disporre di una policy IAM associata che conceda tali autorizzazioni.

Spesso, un team addetto al carico di lavoro vorrà gestire le autorizzazioni richieste dal proprio carico di lavoro. Ciò potrebbe richiedere al team di creare nuovi ruoli IAM e policy di autorizzazione. È possibile acquisire l'ambito massimo delle autorizzazioni che il team può concedere in un limite di autorizzazioni IAM e associare questo documento a un ruolo IAM che il team può quindi utilizzare per gestire i propri ruoli e autorizzazioni IAM. Questo approccio può concedere la libertà di completare il proprio lavoro mitigando al contempo i rischi di accesso amministrativo IAM.

Un passaggio più granulare consiste nell'implementazione delle tecniche di gestione degli accessi privilegiati (PAM) e di gestione temporanea degli accessi elevati (TEAM). Un esempio di PAM consiste nel richiedere ai principali di eseguire l'autenticazione a più fattori prima di intraprendere azioni privilegiate. Per ulteriori informazioni, consulta Configuring MFA-protected API access. TEAM richiede una soluzione che gestisca l'approvazione e i tempi in cui un principale può avere un accesso elevato. Un approccio consiste nell'aggiungere temporaneamente il principale alla policy di attendibilità dei ruoli per un ruolo IAM con accesso elevato. Un altro approccio consiste, in condizioni normali, nel limitare le autorizzazioni concesse a un principale da un ruolo IAM utilizzando una policy di sessione e quindi revocare temporaneamente questa restrizione durante la finestra temporale approvata. Per saperne di più sulle soluzioni convalidate AWS e su alcuni partner selezionati, vedi Temporary elevated access.

Passaggi dell'implementazione

1. Isola i carichi di lavoro e gli ambienti in Account AWS separati.
2. Usa le SCP per ridurre il set massimo di autorizzazioni che possono essere concesse ai principali all'interno degli account membri della tua organizzazione.
   a. Per scrivere le tue SCP, ti consigliamo di adottare un approccio basato sulla lista consentita, che neghi tutte le azioni tranne quelle consentite e le condizioni alle quali sono consentite. Inizia definendo le risorse che desideri controllare e imposta l'effetto su Deny. Utilizza l'elemento NotAction per negare tutte le azioni tranne quelle specificate. Combinalo con una condizione NotLike per definire quando queste azioni sono consentite, se applicabile, come StringNotLike e ArnNotLike.
   b. Vedi Service control policy examples.

3. Utilizza le policy relative alle risorse IAM per definire l'ambito e specificare le condizioni per le azioni consentite sulle risorse. Utilizza le condizioni nelle policy di fiducia dei ruoli IAM per creare restrizioni all'assunzione dei ruoli.
4. Assegna limiti di autorizzazione IAM ai ruoli IAM che i team del carico di lavoro possono quindi utilizzare per gestire i propri ruoli e autorizzazioni IAM.
5. Valuta le soluzioni PAM e TEAM in base alle tue esigenze.

Risorse

Documenti correlati:

- Data perimeters on AWS
- Establish permissions guardrails using data perimeters
- Policy evaluation logic

Esempi correlati:

- Service control policy examples

Strumenti correlati:

- AWS Solution: Temporary Elevated Access Management
- Validated security partner solutions for TEAM

SEC03-BP06 Gestione degli accessi in base al ciclo di vita

Monitora e regola le autorizzazioni concesse ai tuoi principali (utenti, ruoli e gruppi) durante il loro ciclo di vita all'interno dell'organizzazione. Adatta le appartenenze ai gruppi quando gli utenti cambiano ruolo e rimuovi l'accesso quando un utente lascia l'organizzazione.

Risultato desiderato: puoi monitorare e regolare le autorizzazioni durante l'intero ciclo di vita dei principali all'interno dell'organizzazione, riducendo il rischio di privilegi non necessari. Concedi l'accesso appropriato quando crei un utente. L'accesso viene modificato man mano che cambiano le responsabilità dell'utente e lo si rimuove quando l'utente non è più attivo o ha lasciato l'organizzazione. Gestisci centralmente le modifiche ai tuoi utenti, ruoli e gruppi. Utilizza l'automazione per propagare le modifiche agli ambienti AWS.

Anti-pattern comuni:

- Concedi alle identità privilegi di accesso eccessivi o estesi, al di là di quanto richiesto inizialmente.
- I privilegi di accesso non vengono rivisti e modificati poiché i ruoli e le responsabilità delle identità cambiano nel tempo.
- Le identità inattive o terminate vengono lasciate con privilegi di accesso attivi. Ciò aumenta il rischio di accessi non autorizzati.
- La gestione del ciclo di vita dell'identità non viene automatizzata.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Gestisci e adatta attentamente i privilegi di accesso che concedi alle identità (come utenti, ruoli, gruppi) durante il loro ciclo di vita. Questo ciclo di vita include la fase iniziale di onboarding, i continui cambiamenti di ruoli e responsabilità e l'eventuale offboarding o cessazione. Gestisci in modo proattivo l'accesso in base alla fase del ciclo di vita per mantenere il livello di accesso appropriato. Resta conforme al principio del privilegio minimo per ridurre il rischio di privilegi di accesso eccessivi o non necessari.

Puoi gestire il ciclo di vita di IAM users direttamente all'interno di Account AWS o tramite federazione dal tuo fornitore di identità della forza lavoro a AWS IAM Identity Center. Per IAM users è possibile creare, modificare ed eliminare gli utenti e le relative autorizzazioni associate in Account AWS. Per gli utenti federati, puoi utilizzare IAM Identity Center per gestire il loro ciclo di vita sincronizzando le informazioni sugli utenti e sui gruppi dal provider di identità dell'organizzazione mediante il protocollo System for Cross-domain Identity Management (SCIM).

SCIM è un protocollo standard aperto per il provisioning e il deprovisioning automatici delle identità degli utenti su diversi sistemi. Integrando il tuo provider di identità con IAM Identity Center tramite SCIM, puoi sincronizzare automaticamente le informazioni sugli utenti e sui gruppi, verificando che i privilegi di accesso siano concessi, modificati o revocati in base ai cambiamenti nella fonte di identità autorevole dell'organizzazione.

Man mano che i ruoli e le responsabilità dei dipendenti cambiano all'interno dell'organizzazione, modifica di conseguenza i loro privilegi di accesso. È possibile utilizzare i set di autorizzazioni di IAM Identity Center per definire diversi ruoli o responsabilità lavorative e associarli alle policy IAM e alle autorizzazioni appropriate. Quando il ruolo di un dipendente cambia, puoi aggiornare il set di

autorizzazioni assegnato per riflettere le nuove responsabilità. Verifica che il dipendente disponga dell'accesso necessario rispettando il principio del privilegio minimo.

Passaggi dell'implementazione

1. Definisci e documenta un processo del ciclo di vita della gestione degli accessi, comprese le procedure per la concessione dell'accesso iniziale, le revisioni periodiche e l'offboarding.
2. Implementa ruoli, gruppi e limiti di autorizzazioni IAM per gestire l'accesso collettivamente e applicare i livelli di accesso massimi consentiti.
3. Effettua l'integrazione con un provider di identità federato (come Microsoft Active Directory, Okta, Ping Identity) come fonte autorevole per le informazioni sugli utenti e sui gruppi utilizzando IAM Identity Center.
4. Utilizza il protocollo SCIM per sincronizzare le informazioni su utenti e gruppi dal provider di identità nell'Identity Store di IAM Identity Center.
5. Crea set di autorizzazioni in IAM Identity Center che rappresentino diversi ruoli o responsabilità all'interno della tua organizzazione. Definisci le policy e le autorizzazioni IAM appropriate per ogni set di autorizzazioni.
6. Implementa revisioni regolari degli accessi, la loro revoca tempestiva e il miglioramento continuo del processo del ciclo di vita della gestione degli accessi.
7. Fornisci formazione e sensibilizzazione ai dipendenti sulle best practice di gestione degli accessi.

Risorse

Best practice correlate:

- SEC02-BP04 Fai affidamento su un provider di identità centralizzato

Documenti correlati:

- Manage your identity source
- Manage identities in IAM Identity Center
- Using AWS Identity and Access Management Access Analyzer
- IAM Access Analyzer policy generation

Video correlati:

- AWS re:Inforce 2023 - Manage temporary elevated access with AWS IAM Identity Center
- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center
- AWS re:Invent 2022 - Harness power of IAM policies & rein in permissions w/Access Analyzer

SEC03-BP07 Analisi dell'accesso pubblico e multi-account

Monitora continuamente i risultati che evidenziano l'accesso pubblico e multi-account. Limita l'accesso pubblico e multi-account alle risorse che lo richiedono.

Risultato desiderato: sapere quali risorse AWS sono condivise e con chi. Monitora e sottoponi costantemente a audit le risorse condivise per verificare che siano condivise solo con i principali autorizzati.

Anti-pattern comuni:

- Assenza di un inventario delle risorse condivise.
- Mancanza di un processo di approvazione dell'accesso multi-account e dell'accesso pubblico alle risorse.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Se l'account è in AWS Organizations, puoi concedere l'accesso alle risorse all'intera organizzazione, a specifiche unità organizzative o a singoli account. Se l'account non è membro di un'organizzazione, puoi condividere le risorse con account individuali. Puoi concedere l'accesso multi-account diretto utilizzando policy collegate a risorse, ad esempio policy di bucket Amazon Simple Storage Service (Amazon S3) o consentendo a un principale in un altro account di assumere un ruolo IAM nel tuo account. Quando utilizzi le policy sulle risorse, verifica che l'accesso sia concesso solo ai principali autorizzati. Definisci un processo per approvare tutte le risorse che devono essere pubblicamente disponibili.

AWS Identity and Access Management Access Analyzer utilizza la sicurezza comprovabile per identificare tutti i percorsi di accesso a una risorsa dall'esterno del suo account. Esamina continuamente le policy delle risorse e segnala i risultati dell'accesso pubblico e multi-account per semplificare l'analisi di accessi potenzialmente estesi. Considera di configurare IAM Access Analyzer con AWS Organizations per assicurarti di avere visibilità su tutti gli account. IAM Access Analyzer

consente inoltre di vedere in anteprima i risultati prima di implementare le autorizzazioni della risorsa. Questo consente di convalidare che le modifiche alla policy concedono solo l'accesso multi-account e pubblico autorizzati alle risorse. Quando progetti l'accesso multi-account, puoi utilizzare le policy di attendibilità per controllare in quali casi un ruolo può essere assunto. Ad esempio, puoi utilizzare la chiave di condizione `PrincipalOrgId` per respingere il tentativo di assumere un ruolo al di fuori di AWS Organizations.

AWS Config può segnalare le risorse che non sono configurate correttamente e, attraverso i controlli delle policy AWS Config, può rilevare le risorse con accesso pubblico configurato. Servizi quali AWS Control Tower e AWS Security Hub semplificano l'implementazione dei controlli e guardrail investigativi su AWS Organizations per identificare e correggere le risorse esposte pubblicamente. Ad esempio, AWS Control Tower ha un guardrail gestito in grado di rilevare l'eventuale presenza di snapshot Amazon EBS ripristinabili da Account AWS.

Passaggi dell'implementazione

- Considera di abilitare AWS Config per AWS Organizations: AWS Config consente di aggregare i risultati di più account all'interno di un AWS Organizations a un account amministratore delegato. In questo modo si ottiene una visione completa che consente di implementare Regole di AWS Config su più account per rilevare le risorse accessibili pubblicamente.
- Configura AWS Identity and Access Management Access Analyzer. IAM Access Analyzer ti aiuta a identificare le risorse nell'organizzazione e negli account, come ad esempio bucket Amazon S3 o ruoli IAMche sono condivisi con un'entità esterna.
- Utilizza la riparazione automatica in AWS Config per rispondere alle modifiche della configurazione di accesso pubblico dei bucket Amazon S3: puoi riattivare automaticamente le impostazioni di blocco dell'accesso pubblico per i bucket Amazon S3.
- Implementa il monitoraggio e gli avvisi per stabilire se i bucket Amazon S3 sono diventati pubblici: devi disporre di monitoraggio e avvisi per stabilire quando Amazon S3 Block Public Access è disabilitato e se i bucket Amazon S3 diventano pubblici. Inoltre, se stai utilizzando AWS Organizations, puoi creare una policy di controllo del servizio che impedisce di modificare le policy Amazon S3 di accesso pubblico. AWS Trusted Advisor controlla i bucket Amazon S3 che hanno autorizzazioni di accesso aperte. Le autorizzazioni bucket che concedono, caricano o eliminano l'accesso per chiunque danno origine a potenziali problemi di sicurezza, consentendo a chiunque di aggiungere, modificare o rimuovere elementi in un bucket. Il controllo di Trusted Advisor esamina le autorizzazioni bucket esplicite e le policy associate che possono prevalere sulle autorizzazioni bucket. Puoi anche utilizzare AWS Config per monitorare l'accesso pubblico ai bucket Amazon S3. Per ulteriori informazioni, consulta How to Use AWS Config to Monitor for and Respond to Amazon

S3 Buckets Allowing Public Access (Come utilizzare AWS Config per monitorare e gestire i bucket Amazon S3 che consentono l'accesso pubblico). Durante la revisione degli accessi, è importante considerare quali tipi di dati sono contenuti nei bucket Amazon S3. Amazon Macie aiuta a scoprire e a proteggere i dati sensibili, come PII, PHI, e le credenziali, come le chiavi private o quelle AWS.

Risorse

Documenti correlati:

- Utilizzo di AWS Identity and Access Management Access Analyzer
- AWS Control Tower controls library (Libreria di controlli di AWS Control Tower)
- AWS Foundational Security Best Practices standard (Standard AWS Foundational Security Best Practices)
- AWS Config Managed Rules (Regole gestite di AWS Config)
- Riferimento dei controlli AWS Trusted Advisor
- Monitoraggio dei risultati dei controlli AWS Trusted Advisor con Amazon EventBridge
- Managing AWS Config Rules Across All Accounts in Your Organization (Gestire le regole di configurazione AWS tra tutti gli account dell'organizzazione)
- AWS Config e AWS Organizations

Video correlati:

- Best Practices for securing your multi-account environment (Best practice per la protezione dell'ambiente multi-account)
- Dive Deep into IAM Access Analyzer (Approfondire l'analisi degli accessi IAM)

SEC03-BP08 Condivisione delle risorse in modo sicuro all'interno dell'organizzazione

Con l'aumento del numero di carichi di lavoro, è possibile che sia necessario condividere l'accesso alle risorse in tali carichi di lavoro o eseguire il provisioning delle risorse più volte su più account. Possono esistere costrutti per segmentare il proprio l'ambiente, come ad esempio ambienti di sviluppo, di test e di produzione. Tuttavia, la presenza di costrutti di separazione non limita la possibilità di condividere in modo sicuro. La condivisione di componenti che si sovrappongono consente di ridurre i costi operativi e di garantire un'esperienza coerente, senza dover intuire cosa potrebbe sfuggire durante la creazione della stessa risorsa più volte.

Risultato desiderato: ridurre al minimo gli accessi indesiderati utilizzando metodi sicuri per condividere le risorse all'interno dell'organizzazione e contribuire all'iniziativa di prevenzione della perdita di dati. Ridurre i costi operativi rispetto alla gestione dei singoli componenti, ridurre gli errori dovuti alla creazione manuale dello stesso componente più volte e aumentare la scalabilità dei carichi di lavoro. I tempi di risoluzione in caso di guasti multipli sono ridotti e la sicurezza nel determinare quando un componente non è più necessario è aumentata. Per una guida prescrittiva sull'analisi delle risorse condivise dall'esterno, consulta SEC03-BP07 Analisi dell'accesso pubblico e multi-account.

Anti-pattern comuni:

- Mancanza di un processo per il monitoraggio continuo e segnalazione automatica di azioni esterne inaspettate.
- Mancanza di una linea di base su ciò che deve e ciò che non deve essere condiviso.
- Scelta di una policy di ampia apertura piuttosto che di una condivisione esplicita quando richiesto.
- Creazione manuale di risorse fondamentali che si sovrappongono quando necessario.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Progetta i controlli e i modelli di accesso per gestire il consumo di risorse condivise in modo sicuro e solo con entità fidate. Monitora le risorse condivise e controllane costantemente l'accesso ricevendo un avviso in caso di condivisione inappropriata o inaspettata. Consulta Analisi dell'accesso pubblico e multi-account come supporto per stabilire una governance che riduca l'accesso esterno alle sole risorse che lo richiedono e per definire un processo di monitoraggio continuo e di avviso automatico.

La condivisione multi-account in AWS Organizations è supportata da una serie di servizi AWS, come AWS Security Hub, Amazon GuardDuty e AWS Backup. Questi servizi permettono di condividere i dati con un account centrale, di accedere a un account centrale o di gestire risorse e dati da un account centrale. Ad esempio, AWS Security Hub può trasferire i risultati dai singoli account a un account centrale in cui è possibile visualizzare tutti i risultati. AWS Backup può eseguire un backup di una risorsa e condividerlo tra gli account. Puoi utilizzare AWS Resource Access Manager (AWS RAM) per condividere altre risorse comuni, quali sottoreti VPC e allegati Transit Gateway, AWS Network Firewall o pipeline Amazon SageMaker.

Per limitare l'account alla condivisione di risorse solo all'interno dell'organizzazione, utilizza le policy di controllo dei servizi (Service Control Policies, SCP) per impedire l'accesso ai principali esterni. Quando condividi le risorse, combina controlli basati sull'identità e controlli di rete per creare un

perimetro di dati per l'organizzazione in modo da proteggere dall'accesso non intenzionale. Un perimetro di dati è un insieme di guardrail preventivi che aiutano a verificare che solo le identità fidate accedano a risorse fidate dalle reti previste. Questi controlli pongono limiti adeguati alle risorse che possono essere condivise e impediscono la condivisione o l'esposizione di risorse che non sono consentite. Ad esempio, nell'ambito del perimetro dei dati, è possibile utilizzare le policy degli endpoint VPC e la condizione `AWS:PrincipalOrgId` per assicurarsi che le identità che accedono ai bucket Amazon S3 appartengano alla propria organizzazione. È importante notare che le policy di controllo dei servizi non si applicano ai ruoli correlati ai servizi (Service-Linked Roles, LSR) o ai principali del servizio AWS.

Quando utilizzi Amazon S3, disabilita le ACL per il bucket Amazon S3 e utilizza le policy IAM per definire il controllo degli accessi. Per delimitare un accesso a un'origine Amazon S3 da Amazon CloudFront, migra dall'identità di accesso origine (OAI) al controllo di accesso origine (OAC) che supporta funzionalità aggiuntive, inclusa la crittografia lato server con AWS Key Management Service.

In alcuni casi, può essere necessario condividere le risorse al di fuori dell'organizzazione o concedere a terzi l'accesso alle risorse stesse. Per una guida prescrittiva sulla gestione delle autorizzazioni per la condivisione di risorse all'esterno, consulta Gestione delle autorizzazioni.

Passaggi dell'implementazione

1. Utilizzo di AWS Organizations.

   AWS Organizations è un servizio di gestione degli account che consente di consolidare più Account AWS in un'organizzazione creata e gestita centralmente. È possibile raggruppare gli account in unità organizzative (OU) e associare policy diverse a ciascuna di esse per soddisfare le esigenze di bilancio, sicurezza e conformità. È inoltre possibile controllare il modo in cui i servizi di Intelligenza Artificiale (IA) e di machine learning (ML) di AWS possono raccogliere e archiviare i dati e utilizzare la gestione multi-account dei servizi AWS integrati nelle Organizations.

2. Integrazione delle AWS Organizations con i servizi AWS.

   Quando si abilita un servizio AWS a svolgere attività per conto dell'utente negli account membri dell'organizzazione, AWS Organizations crea un ruolo IAM collegato al servizio in ogni account membro. L'accesso attendibile deve essere gestito tramite la AWS Management Console, le API AWS o la AWS CLI. Per una guida prescrittiva sull'abilitazione dell'accesso attendibile, consulta Uso di AWS Organizations con altri servizi AWS e Servizi AWS che puoi utilizzare con Organizations.

3. Definizione di un perimetro di dati.

   Il perimetro AWS è tipicamente rappresentato come un'organizzazione gestita da AWS Organizations. Insieme alle reti e ai sistemi on-premise, l'accesso alle risorse AWS è ciò che molti considerano il perimetro di My AWS. L'obiettivo del perimetro è verificare che l'accesso sia consentito se l'identità è attendibile, la risorsa è attendibile e la rete è conforme.

   a. Definizione e implementazione dei perimetri.

      Segui i passaggi descritti in Perimeter implementation (Implementazione del perimetro) nel whitepaper Building a Perimeter on AWS (Costruire un perimetro su AWS) per qualsiasi condizione di autorizzazione. Per una guida prescrittiva sulla protezione del livello di rete, consulta Protezione delle reti.

   b. Monitoraggio e segnalazione continui.

      AWS Identity and Access Management Access Analyzer aiuta a identificare le risorse dell'organizzazione e gli account condivisi con entità esterne. Puoi integrare IAM Access Analyzer con AWS Security Hub per inviare e aggregare i risultati di una risorsa da IAM Access Analyzer a Security Hub per analizzare la sicurezza dell'ambiente. Per abilitare l'integrazione, abilita sia IAM Access Analyzer che Security Hub in ogni Regione per ogni account. Puoi anche utilizzare Regole di AWS Config per eseguire l'audit della configurazione e avvisare la parte interessata mediante AWS Chatbot con AWS Security Hub. Puoi quindi utilizzare i Documenti di AWS Systems Manager per adottare i provvedimenti correttivi alle risorse non conformi.

   c. Per una guida prescrittiva sul monitoraggio e sull'avviso continuo delle risorse condivise esternamente, consulta Analisi dell'accesso pubblico e multi-account.

4. Utilizza la condivisione delle risorse nei servizi AWS e delimitale di conseguenza.

   Molti servizi AWS consentono di condividere le risorse con un altro account o di puntare a una risorsa di un altro account, come Amazon Machine Image (AMI) e AWS Resource Access Manager (AWS RAM). Delimita l'API `ModifyImageAttribute` per specificare gli account affidabili con cui condividere l'AMI. Specifica la condizione `ram:RequestedAllowsExternalPrincipals` quando si utilizza AWS RAM per limitare la condivisione solo alla propria organizzazione, per evitare l'accesso da parte di identità non affidabili. Per indicazioni e considerazioni prescrittive Resource sharing and external targets (Condivisione delle risorse e target esterni).

5. Utilizzare AWS RAM per condivisioni sicure con un account o con altri Account AWS.

AWS RAM consente di condividere in modo sicuro le risorse create con i ruoli e gli utenti del proprio account e con altri utenti Account AWS. In un ambiente multi-account, AWS RAM consente di creare una risorsa una sola volta e di condividerla con altri account. Questo approccio contribuisce a ridurre i costi operativi, fornendo al contempo coerenza, visibilità e verificabilità grazie alle integrazioni con Amazon CloudWatch e AWS CloudTrail, che non si ottengono quando si utilizza l'accesso multi-account.

Se si dispone di risorse condivise in precedenza utilizzando una policy basata sulle risorse, è possibile utilizzare l'API `PromoteResourceShareCreatedFromPolicy` o un'API equivalente per promuovere il passaggio da una condivisione di risorse a una condivisione completa di risorse AWS RAM.

In alcuni casi, potrebbe essere necessario adottare ulteriori misure per condividere le risorse. Ad esempio, per condividere un'istantanea crittografata è necessario condividere una chiave AWS KMS.

Risorse

Best practice correlate:

- SEC03-BP07 Analisi dell'accesso pubblico e multi-account
- SEC03-BP09 Condivisione sicura delle risorse con terze parti
- SEC05-BP01 Creazione di livelli di rete

Documenti correlati:

- Il proprietario del bucket concede autorizzazioni multi-account per gli oggetti che non sono di sua proprietà
- How to use Trust Policies with IAM (Come utilizzare le policy di attendibilità con IAM)
- Building Data Perimeter on AWS (Creazione del perimetro dei dati in AWS)
- How to use an external ID when granting a third party access to your AWS resources (Come utilizzare un ID esterno quando si concede a una terza parte l'accesso alle risorse AWS)
- Servizi AWS che puoi utilizzare con AWS Organizations
- Establishing a data perimeter on AWS: Allow only trusted identities to access company data (Applicazione di un perimetro dei dati in AWS: consentire l'accesso ai dati aziendali solo alle identità attendibili)

Video correlati:

- Granular Access with AWS Resource Access Manager (Accesso granulare con Gestione degli accessi alle risorse AWS)
- Securing your data perimeter with VPC endpoints (Protezione del perimetro dei dati con gli endpoint VPC)
- Establishing a data perimeter on AWS(Applicazione di un perimetro dei dati in AWS)

Strumenti correlati:

- Esempi di policy sul perimetro dei dati

SEC03-BP09 Condivisione sicura delle risorse con terze parti

La sicurezza dell'ambiente cloud non si ferma alla tua organizzazione. L'organizzazione potrebbe affidare a terzi la gestione di una parte dei dati. La gestione dei permessi per il sistema gestito da terzi deve seguire la pratica dell'accesso just-in-time utilizzando il principio del privilegio minimo con credenziali temporanee. Lavorando a stretto contatto con una terza parte, puoi ridurre congiuntamente la portata dell'impatto e il rischio di accesso non intenzionale.

Risultato desiderato: le credenziali AWS Identity and Access Management (IAM) a lungo termine, le chiavi di accesso IAM e le chiavi segrete associate a un utente possono essere utilizzate da chiunque, purché le credenziali siano valide e attive. L'utilizzo di credenziali temporanee e di un ruolo IAM consente di migliorare la sicurezza generale riducendo l'impegno per la manutenzione delle credenziali a lungo termine, compresi i costi di gestione e di funzionamento di questi dati sensibili. Utilizzando un identificatore univoco universale (UUID) per l'ID esterno nella policy di attendibilità IAM e mantenendo sotto il proprio controllo le policy IAM collegate al ruolo IAM, puoi sottoporre a audit e verificare che l'accesso concesso a terzi non sia troppo permissivo. Per una guida prescrittiva sull'analisi delle risorse condivise dall'esterno, consulta SEC03-BP07 Analisi dell'accesso pubblico e multi-account.

Anti-pattern comuni:

- Utilizzo della policy di attendibilità IAM predefinita senza alcuna condizione.
- Utilizzo di credenziali IAM e chiavi di accesso a lungo termine.
- Riutilizzo di ID esterni.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

In alcuni casi, può essere necessario condividere le risorse al di fuori di AWS Organizations o concedere a terzi l'accesso alle risorse stesse. Ad esempio, una terza parte potrebbe fornire una soluzione di monitoraggio che necessita di accedere alle risorse del tuo account. In questi casi, devi creare un ruolo IAM multi-account con i soli privilegi necessari alla terza parte. Inoltre, devi definire una policy di attendibilità utilizzando la condizione di ID esterno. L'utilizzo di un ID esterno da parte tua o della terza parte può comportare la generazione di un ID univoco per ogni cliente, terza parte o tenancy. Una volta creato, l'ID univoco non deve essere controllato da nessuno, se non da te. La terza parte deve implementare un processo per collegare l'ID esterno al cliente in modo sicuro, verificabile e riproducibile.

Puoi anche utilizzare IAM Roles Anywhere per gestire ruoli IAM per le applicazioni esterne ad AWS che utilizzano le API AWS.

Se la terza parte non ha più bisogno di accedere al tuo ambiente, rimuovi il ruolo. Evita di fornire a terze parti credenziali a lungo termine. Mantieni la visibilità degli altri servizi AWS che supportano la condivisione. Ad esempio, AWS Well-Architected Tool consente la condivisione di un carico di lavoro con altri Account AWS e AWS Resource Access Manager ti aiuta a condividere in modo sicuro una risorsa AWS di tua proprietà con altri account.

Passaggi dell'implementazione

1. Utilizzare i ruoli multi-account per fornire l'accesso agli account esterni.

   I ruoli multi-account riducono la quantità di informazioni sensibili archiviate da account esterni e da terze parti per l'assistenza ai propri clienti. I ruoli multi-account consentono di concedere l'accesso alle risorse AWS del proprio account in modo sicuro a terzi, come i AWS Partner o altri account dell'organizzazione, mantenendo la possibilità di gestire e sottoporre a audit tale accesso.

   La terza parte può fornire il servizio da un'infrastruttura ibrida o, in alternativa, estrarre i dati in una sede esterna. IAM Roles Anywhere consente ai carichi di lavoro di terze parti di interagire in modo sicuro con i carichi di lavoro AWS e di ridurre ulteriormente la necessità di credenziali a lungo termine.

   Non devi utilizzare credenziali a lungo termine o chiavi di accesso associate agli utenti per fornire accesso ad account esterni. Per fornire l'accesso multi-account invece, occorre utilizzare i ruoli multi-account.

2. Utilizzare un ID esterno con terze parti.

   L'utilizzo di un ID esterno consente di designare chi può assumere un ruolo in una policy di attendibilità IAM. La policy di attendibilità può richiedere che l'utente che assume il ruolo dichiari la condizione e l'obiettivo in cui sta operando. Inoltre, il proprietario dell'account può consentire l'assunzione del ruolo solo in determinate circostanze. La funzione principale dell'ID esterno è quella di affrontare e prevenire il problema del confused deputy.

   Utilizza un ID esterno se sei il proprietario di un Account AWS e hai configurato un ruolo per una terza parte che accede ad altri Account AWS oltre al tuo, oppure quando ti trovi nella posizione di assumere ruoli per conto di diversi clienti. Collabora con la terza parte o con il AWS Partner per stabilire una condizione di ID esterno da includere nelle policy di attendibilità IAM.

3. Utilizzare ID esterni universalmente univoci.

   Implementa un processo che generi un valore univoco casuale per un ID esterno, ad esempio un identificatore univoco universale (UUID). Una terza parte che riutilizza gli ID esterni tra diversi clienti non risolve il problema del confused deputy, perché il cliente A potrebbe essere in grado di visualizzare i dati del cliente B utilizzando l'ARN del ruolo del cliente B insieme all'ID esterno duplicato. In un ambiente multi-tenant, in cui una terza parte supporta più clienti con diversi Account AWS, la terza parte deve utilizzare un ID univoco diverso come ID esterno per ogni Account AWS. La terza parte è responsabile del rilevamento di ID esterni duplicati e della mappatura sicura di ciascun cliente al rispettivo ID esterno. La terza parte deve verificare di poter assumere il ruolo solo quando specifica l'ID esterno. La terza parte deve astenersi dal memorizzare l'ARN del ruolo del cliente e l'ID esterno fino a quando non è richiesto l'ID esterno.

   L'ID esterno non viene trattato come un segreto, ma non deve essere un valore facilmente individuabile, come un numero di telefono, un nome o un ID di account. Rendi l'ID esterno un campo di sola lettura, in modo che non possa essere modificato per rappresentare la configurazione.

   L'ID esterno può essere generato da te o dalla terza parte. Definisci un processo per stabilire chi è responsabile della generazione dell'ID. Indipendentemente dall'entità che crea l'ID esterno, la terza parte fa rispettare l'unicità e i formati in modo coerente tra i clienti.

4. Rendere obsolete le credenziali a lungo termine fornite dal cliente.

   Rendi obsoleto l'uso di credenziali a lungo termine e utilizza ruoli multi-account oppure IAM Roles Anywhere. Se devi utilizzare credenziali a lungo termine, stabilisci un piano per migrare verso l'accesso basato sui ruoli. Per dettagli sulla gestione delle chiavi, consulta Identity Management

(Gestione dell'identità). Collaborare inoltre con il team dell'Account AWS e con la terza parte per stabilire un runbook di mitigazione dei rischi. Per una guida prescrittiva sulla risposta e la mitigazione dell'impatto potenziale di un incidente di sicurezza, consulta Incident response (Risposta agli incidenti).

5. Verifica che l'impostazione abbia una guida prescrittiva o sia automatizzata.

   La policy creata per l'accesso multi-account deve seguire il principio del privilegio minimo. La terza parte deve fornire un documento sulla policy del ruolo o un meccanismo di configurazione automatica che utilizzi un modello AWS CloudFormation o un equivalente per l'utente. In questo modo si riduce la possibilità di errori associati alla creazione manuale della policy e si offre una traccia verificabile. Per ulteriori informazioni sull'utilizzo di un modello AWS CloudFormation per creare ruoli trasversali agli account, consulta Cross-Account Roles (Ruoli multi-account).

   La terza parte deve fornire un meccanismo di configurazione automatizzato e verificabile. Tuttavia, utilizzando il documento della policy sui ruoli che delinea gli accessi necessari, è possibile automatizzare l'impostazione del ruolo. Con un modello AWS CloudFormation o equivalente, è necessario monitorare le modifiche con il rilevamento delle derive come parte della pratica di audit.

6. Account per le modifiche.

   La struttura del tuo account, la tua necessità di una terza parte o l'offerta di servizi che ti viene fornita possono cambiare. Occorre anticipare i cambiamenti e i fallimenti e pianificare di conseguenza con le persone, i processi e le tecnologie adeguati. Sottoponi periodicamente a audit il livello di accesso fornito e implementa metodi di rilevamento per avvisare l'utente di cambiamenti inattesi. Monitora e sottoponi a audit l'uso del ruolo e del datastore degli ID esterni. Devi essere pronto a revocare l'accesso a terzi, in modo temporaneo o permanente, in seguito a modifiche o modelli di accesso imprevisti. Inoltre, valuta l'impatto dell'operazione di revoca, compreso il tempo necessario per eseguirla, le persone coinvolte, il costo e l'impatto su altre risorse.

   Per una guida prescrittiva sui metodi di rilevamento, consulta Best practice di rilevamento.

Risorse

Best practice correlate:

- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC03-BP05 Definizione dei guardrail per le autorizzazioni dell'organizzazione
- SEC03-BP06 Gestione degli accessi in base al ciclo di vita

- SEC03-BP07 Analisi dell'accesso pubblico e multi-account
- SEC04 Rilevamento

Documenti correlati:

- Il proprietario del bucket concede autorizzazioni multi-account per gli oggetti che non sono di sua proprietà
- How to use trust policies with IAM roles (Come utilizzare le policy di attendibilità con i ruoli IAM)
- Delega dell'accesso tra Account AWS tramite i ruoli IAM
- How do I access resources in another Account AWS using IAM? (Come faccio ad accedere alle risorse di un altro account AWS utilizzando IAM?)
- Best practice per la sicurezza in IAM
- Logica di valutazione della policy multiaccount
- Come utilizzare un ID esterno quando si concede a una terza parte l'accesso alle proprie risorse AWS
- Collecting Information from AWS CloudFormation Resources Created in External Accounts with Custom Resources (Raccolta di informazioni dalle risorse AWS CloudFormation create in account esterni con risorse personalizzate)
- Securely Using External ID for Accessing AWS Accounts Owned by Others (Utilizzo sicuro dell'ID esterno per l'accesso agli account AWS di proprietà di altri)
- Extend IAM roles to workloads outside of IAM with IAM Roles Anywhere (Estendere i ruoli IAM a carichi di lavoro esterni a IAM con IAM Roles Anywhere)

Video correlati:

- How do I allow users or roles in a separate Account AWS access to my Account AWS? (Come posso consentire agli utenti o ai ruoli di un account AWS separato di accedere al mio account AWS?)
- AWS re:Invent 2018: Become an IAM Policy Master in 60 Minutes or Less (Diventa un IAM Policy Master in 60 minuti)
- AWS Knowledge Center Live: IAM Best Practices and Design Decisions (Knowledge Center AWS in diretta: best practice e decisioni di progettazione IAM)

Esempi correlati:

- Well-Architected Lab - Lambda cross account IAM role assumption (Level 300) [Well-Architected Lab: Assunzione di ruoli IAM per account incrociati Lambda (livello 300)]
- Configure cross-account access to Amazon DynamoDB (Configurare l'accesso multi-account ad Amazon DynamoDB)
- AWS STS Network Query Tool (Strumento di consultazione della rete AWS STS)

Rilevamento

Domanda

- SEC 4. In che modo individui ed esamini gli eventi di sicurezza?

SEC 4. In che modo individui ed esamini gli eventi di sicurezza?

Acquisisci ed analizza gli eventi a partire da log e parametri per acquistare visibilità. Agisci su eventi di sicurezza e potenziali minacce per contribuire a rendere sicuro il carico di lavoro.

Best practice

- SEC04-BP01 Configurazione dei registri di servizi e applicazioni
- SEC04-BP02 Acquisizione di log, esiti e metriche in posizioni standardizzate.
- SEC04-BP03 Correlazione e arricchimento degli avvisi di sicurezza
- SEC04-BP04 Avvio della riparazione delle risorse non conformi

SEC04-BP01 Configurazione dei registri di servizi e applicazioni

Mantieni i log degli eventi di sicurezza dei servizi e delle applicazioni. Si tratta di un principio fondamentale di sicurezza per i casi d'uso di audit, indagini e operazioni, nonché di un requisito di sicurezza comune guidato da standard, policy e procedure di governance, rischio e conformità (GRC).

Risultato desiderato: un'organizzazione deve essere in grado di recuperare in modo affidabile e coerente i log degli eventi di sicurezza dei servizi e delle applicazioni AWS in modo tempestivo, quando è necessario soddisfare un processo o un obbligo interno, come la risposta a un incidente di sicurezza. Considera la possibilità di centralizzare i log per ottenere migliori risultati operativi.

Anti-pattern comuni:

- Log archiviati in modo perpetuo o cancellati troppo presto.
- Tutti possono accedere ai log.
- Affidarsi interamente a processi manuali per la governance e l'utilizzo dei log.
- Archiviazione di ogni singolo tipo di log nel caso in cui sia necessario.
- Controllo dell'integrità del log solo quando è necessario.

Vantaggi della definizione di questa best practice: implementare un meccanismo di root cause analysis (RCA) per gli incidenti di sicurezza e una fonte di prove per gli obblighi di governance, rischio e conformità.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Durante un'indagine di sicurezza o in altri casi d'uso basati sui tuoi requisiti, devi essere in grado di esaminare i log pertinenti per registrare e comprendere l'intera portata e la tempistica dell'incidente. I log sono necessari anche per la generazione di avvisi, che indicano che sono avvenute determinate azioni di interesse. È fondamentale selezionare, attivare, memorizzare e impostare i meccanismi di interrogazione e recupero e gli avvisi.

Passaggi dell'implementazione

- Selezionare e abilitare le origini dei log. Prima di un'indagine di sicurezza, devi acquisire i log rilevanti per ricostruire retroattivamente l'attività in un Account AWS. Seleziona e attiva le origini dei log rilevanti per i carichi di lavoro.

  I criteri di selezione delle origini dei log devono essere basati sui casi d'uso richiesti dall'azienda. Stabilisci un percorso per ogni Account AWS utilizzando AWS CloudTrail o un percorso AWS Organizations e configura per esso un bucket Amazon S3.

  AWS CloudTrail è un servizio di registrazione che tiene traccia delle chiamate API effettuate su un Account AWS, catturando l'attività del servizio AWS. È abilitato per impostazione predefinita e prevede una conservazione di 90 giorni degli eventi di gestione che possono essere recuperati attraverso la cronologia degli eventi CloudTrail utilizzando la AWS Management Console, la AWS CLI o un AWS SDK. Per una conservazione e una visibilità più lunghe degli eventi di dati, crea un percorso CloudTrail e associalo a un bucket Amazon S3 e, facoltativamente, a un gruppo di log Amazon CloudWatch. In alternativa, puoi creare un CloudTrail Lake, che mantiene i log di CloudTrail per un massimo di sette anni e fornisce una funzionalità di query basata su SQL

AWS consiglia ai clienti che utilizzano un VPC di abilitare i log del traffico di rete e del DNS utilizzando rispettivamente i log di flusso VPC e i log delle query del resolver Amazon Route 53 e di inviarli in streaming a un bucket Amazon S3 o a un gruppo di log CloudWatch. Il log di flusso VPC può essere creato per un VPC, una sottorete o un'interfaccia di rete. Per i log di flusso VPC, puoi scegliere come e dove utilizzarli per ridurre i costi.

I log AWS CloudTrail, i log di flusso VPC e i log delle query del resolver Route 53 sono le origini dei log di base per supportare le indagini sulla sicurezza in AWS. Puoi anche utilizzare Amazon Security Lake per raccogliere, normalizzare e archiviare questi dati di log in formato Apache Parquet e Open Cybersecurity Schema Framework (OCSF), pronti per essere interrogati. Security Lake supporta anche altri log AWS e log provenienti da origini di terze parti.

I servizi AWS possono generare log non acquisiti dalle origini di log di base, come log di Elastic Load Balancing, log di AWS WAF, log di AWS Config, risultati di Amazon GuardDuty, log di audit di Amazon Elastic Kubernetes Service (Amazon EKS) e log del sistema operativo e delle applicazioni delle istanze Amazon EC2. Per un elenco completo delle opzioni di registrazione e monitoraggio, consulta Appendix A: Cloud capability definitions – Logging and Events (Appendice A: Definizioni delle capacità del cloud - Registrazione ed eventi) della AWS Security Incident Response Guide (Guida alla risposta agli incidenti di sicurezza di AWS).

- Ricercare le funzionalità di log per ogni servizio e applicazioneAWS: ogni servizio e applicazione AWS offre opzioni per l'archiviazione dei log, ognuna con capacità di conservazione e ciclo di vita proprie. I due servizi di archiviazione dei log più comuni sono Amazon Simple Storage Service (Amazon S3) e Amazon CloudWatch. Per lunghi periodi di conservazione, è consigliabile utilizzare Amazon S3 per la sua economicità e per la flessibilità del ciclo di vita. Se l'opzione principale di registrazione è Amazon CloudWatch Logs, puoi prendere in considerazione l'archiviazione dei log ad accesso meno frequente su Amazon S3.
- Selezionare l'archiviazione dei log: la scelta dell'archiviazione dei log è generalmente legata allo strumento di query utilizzato, alle capacità di conservazione, alla familiarità e al costo. Le opzioni principali per l'archiviazione dei log sono un bucket Amazon S3 o un gruppo CloudWatch Log.

  Un bucket Amazon S3 offre a possibilità di un'archiviazione economica e duratura, con una policy opzionale per il ciclo di vita. I log archiviati nei bucket Amazon S3 possono essere interrogati utilizzando servizi come Amazon Athena.

  Un gruppo di log di CloudWatch offre un'archiviazione durevole e una funzione di interrogazione integrata attraverso CloudWatch Logs Insights.

- Identificare la conservazione appropriata dei log: quando utilizzi un bucket Amazon S3 o un gruppo di log CloudWatch per archiviare i log, è necessario stabilire cicli di vita adeguati per ogni origine di log per ottimizzare i costi di archiviazione e recupero. In genere i clienti hanno a disposizione da tre mesi a un anno di log per le query, con una conservazione fino a sette anni. La scelta della disponibilità e della conservazione deve essere in linea con i requisiti di sicurezza e con un insieme di mandati statutari, normativi e aziendali.
- Abilitare la registrazione per ogni servizio e applicazione AWS con policy di conservazione e ciclo di vita adeguate: per ogni servizio o applicazione AWS nell'organizzazione, cerca le indicazioni specifiche per la configurazione della registrazione:
  - Configure AWS CloudTrail Trail (Configurazione di un percorso AWS CloudTrail)
  - Configure VPC Flow Logs (Configurazione di VPC Flow Logs)
  - Configure Amazon GuardDuty Finding Export (Configurazione dell'esportazione di risultati Amazon GuardDuty)
  - Configure AWS Config recording (Configurazione della registrazione di AWS Config)
  - Configure AWS WAF web ACL traffic (Configurazione del traffico ACL web di AWS WAF)
  - Configure AWS Network Firewall network traffic logs (Configurazione dei log del traffico di rete del firewall di rete AWS)
  - Configure Elastic Load Balancing access logs (Configurazione dei log di accesso di Elastic Load Balancing)
  - Configure Amazon Route 53 resolver query logs (Configurazione dei log delle query del resolver di Amazon Route 53)
  - Configure Amazon RDS logs (Configurazione dei log di Amazon RDS)
  - Configure Amazon EKS Control Plane logs (Configurazione dei log del piano di controllo di Amazon EKS)
  - Configure Amazon CloudWatch agent for Amazon EC2 instances and on-premises servers (Configurazione dell'agente Amazon CloudWatch per istanze Amazon EC2 e server on-premise)
- Selezionare e implementare i meccanismi di interrogazione dei log: per le query sui log, puoi utilizzare CloudWatch Logs Insights per i dati archiviati nei gruppi di log di CloudWatch e Amazon Athena e Amazon OpenSearch Service per i dati archiviati in Amazon S3. Inoltre, puoi utilizzare strumenti di interrogazione di terze parti, come un servizio di gestione delle informazioni e degli eventi di sicurezza (SIEM).

  Il processo di selezione di uno strumento di interrogazione dei log deve considerare gli aspetti relativi a persone, processi e tecnologia delle operazioni di sicurezza. Occorre scegliere uno

strumento che soddisfi i requisiti operativi, aziendali e di sicurezza e che sia accessibile e manutenibile a lungo termine. Tieni presente che gli strumenti di interrogazione dei log funzionano in modo ottimale quando il numero di log da analizzare è mantenuto entro i limiti dello strumento. Non è raro avere più strumenti di interrogazione a causa di vincoli tecnici o di costo.

Ad esempio, puoi ricorrere a uno strumento di gestione delle informazioni e degli eventi di sicurezza (SIEM) di terze parti per eseguire query sugli ultimi 90 giorni di dati, ma utilizzare Athena per eseguire query oltre i 90 giorni a causa dei costi di importazione dei log di un SIEM. Indipendentemente dall'implementazione, verifica che il tuo approccio riduca al minimo il numero di strumenti necessari per massimizzare l'efficienza operativa, soprattutto durante le indagini su un evento di sicurezza.

- Utilizzare i log per gli avvisi: AWS fornisce avvisi attraverso diversi servizi di sicurezza:
  - AWS Config monitora e registra le configurazioni delle risorse AWS e consente di automatizzare la valutazione e la correzione delle configurazioni desiderate.
  - Amazon GuardDuty è un servizio di rilevamento delle minacce che monitora costantemente la presenza di attività dannose e di comportamenti non autorizzati per proteggere gli Account AWS e i carichi di lavoro. GuardDuty acquisisce, aggrega e analizza le informazioni provenienti da origini, come ad esempio gestione AWS CloudTrail ed eventi di dati, log DNS, log di flusso VPC e log di audit Amazon EKS. GuardDuty estrae flussi di dati indipendenti direttamente da CloudTrail, log di flusso VPC, log di query DNS ed Amazon EKS. Non è necessario gestire le policy del bucket Amazon S3 o modificare le modalità di raccolta e archiviazione dei log. È comunque consigliabile mantenere questi registri a fini investigativi e di conformità.
  - AWS Security Hub offre un unico luogo che aggrega, organizza e definisce le priorità degli avvisi di sicurezza o delle scoperte provenienti da più servizi AWS e da prodotti opzionali di terze parti, per fornire una visione completa degli avvisi di sicurezza e dello stato di conformità.

  Esistono anche motori di generazione di avvisi personalizzati per gli avvisi di sicurezza non coperti da questi servizi o per gli avvisi specifici relativi al tuo ambiente. Per informazioni sulla creazione di questi avvisi e rilevamenti, consulta Detection (Rilevamento) nella AWS Security Incident Response Guide (Guida alla risposta agli incidenti di sicurezza di AWS).

Risorse

Best practice correlate:

- SEC04-BP02 Acquisizione di log, esiti e metriche in posizioni standardizzate.
- SEC07-BP04 Definizione della gestione del ciclo di vita dei dati scalabili

- SEC10-BP06 Distribuzione anticipata degli strumenti

Documenti correlati:

- AWS Security Incident Response Guide
- Nozioni di base su Amazon Security Lake
- Nozioni di base su: Amazon CloudWatch Logs
- Soluzione dei partner per la sicurezza: registrazione e monitoraggio

Video correlati:

- AWS re:Invent 2022 - Introducing Amazon Security Lake (re:Invent 2022 - Introduzione ad Amazon Security Lake)

Esempi correlati:

- Assisted Log Enabler for AWS(Abilitatore di log assistito per AWS)
- AWS Security Hub Findings Historical Export (Esportazione cronologica dei risultati di AWS Security Hub)

Strumenti correlati:

- Snowflake for Cybersecurity

SEC04-BP02 Acquisizione di log, esiti e metriche in posizioni standardizzate.

I team di sicurezza si basano su log ed esiti per analizzare gli eventi che possono indicare attività non autorizzate o modifiche non intenzionali. Per semplificare questa analisi, acquisisci i log e gli esiti di sicurezza in posizioni standardizzate. Ciò rende disponibili i punti di interesse dei dati per la correlazione e può semplificare le integrazioni degli strumenti.

Risultato desiderato: un approccio standardizzato per raccogliere, analizzare e visualizzare i dati di log, gli esiti e le metriche. I team di sicurezza possono correlare, analizzare e visualizzare in modo efficiente i dati di sicurezza su sistemi diversi per scoprire potenziali eventi di sicurezza e identificare le anomalie. I sistemi di gestione delle informazioni e degli eventi di sicurezza (SIEM)

o altri meccanismi sono integrati per interrogare e analizzare i dati dei log per risposte tempestive, tracciare ed eseguire escalation degli eventi di sicurezza.

Anti-pattern comuni:

- I team hanno e gestiscono in modo indipendente la raccolta di log e metriche che non è coerente con la strategia di registrazione dell'organizzazione.
- I team non dispongono di controlli di accesso adeguati per limitare la visibilità e l'alterazione dei dati raccolti.
- I team non gestiscono i log, gli esiti e le metriche di sicurezza come parte della loro policy di classificazione dei dati.
- I team trascurano i requisiti di sovranità e localizzazione dei dati durante la configurazione delle raccolte di dati.

Vantaggi della definizione di questa best practice: una soluzione di logging standardizzata per raccogliere e interrogare i dati e gli eventi di log migliora gli approfondimenti derivati dalle informazioni in essi contenute. La configurazione di un ciclo di vita automatizzato per i dati di log raccolti può ridurre i costi sostenuti per l'archiviazione dei log. È possibile creare un controllo di accesso granulare per le informazioni di log raccolte, in base alla sensibilità dei dati e ai modelli di accesso richiesti dai team. Puoi integrare strumenti per correlare, visualizzare e ricavare informazioni dai dati.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

La crescita dell'utilizzo di AWS all'interno di un'organizzazione comporta un numero crescente di carichi di lavoro e ambienti distribuiti. Poiché ognuno di questi carichi di lavoro e ambienti genera dati sull'attività al suo interno, l'acquisizione e l'archiviazione di questi dati a livello locale rappresenta una sfida per le operazioni di sicurezza. I team addetti alla sicurezza utilizzano strumenti come i sistemi SIEM (Security Information and Event Management) per raccogliere dati da origini distribuite e sottoporli a flussi di lavoro di correlazione, analisi e risposta. Ciò richiede la gestione di una serie complessa di autorizzazioni per l'accesso alle varie origini dati e un sovraccarico aggiuntivo nel funzionamento dei processi di estrazione, trasformazione e caricamento (ETL).

Per superare queste sfide, valuta la possibilità di aggregare tutte le origini pertinenti di dati dei log di sicurezza in un account Log Archive, come descritto in Organizing Your AWS Environment Using Multiple Accounts. Ciò include tutti i dati relativi alla sicurezza del carico di lavoro e i log generati dai

servizi AWS, ad esempio AWS CloudTrail, AWS WAF, Elastic Load Balancing e Amazon Route 53. L'acquisizione di questi dati in posizioni standardizzate e in un Account AWS separato presenta diversi vantaggi. Questa pratica aiuta a prevenire la manomissione dei log all'interno di ambienti e carichi di lavoro compromessi, fornisce un unico punto di integrazione per strumenti aggiuntivi e offre un modello più semplificato per la configurazione della conservazione e del ciclo di vita dei dati. Valuta gli impatti della sovranità dei dati, degli ambiti di conformità e di altre normative per determinare se sono necessarie più sedi di archiviazione e periodi di conservazione dei dati di sicurezza.

Per facilitare l'acquisizione e la standardizzazione di log ed esiti, valuta Amazon Security Lake nel tuo account Log Archive. È possibile configurare Security Lake per l'acquisizione automatica dei dati da origini comuni, quali CloudTrail, Route 53, Amazon EKS e VPC Flow Logs. Puoi anche configurare AWS Security Hub come origine dati in Security Lake, consentendoti di mettere in correlazione gli esiti di altri servizi AWS, come Amazon GuardDuty e Amazon Inspector, con i tuoi dati di log. Puoi anche utilizzare integrazioni di origini dati di terze parti o configurare origini dati personalizzate. Tutte le integrazioni standardizzano i dati nel formato Open Cybersecurity Schema Framework (OCSF) e vengono archiviate in bucket Amazon S3 come file Parquet, eliminando la necessità di elaborazione ETL.

L'archiviazione dei dati di sicurezza in posizioni standardizzate offre funzionalità di analisi avanzate. AWS consiglia di distribuire strumenti per l'analisi della sicurezza che operano in un ambiente AWS in un account Security Tooling separato dall'account Log Archive. Questo approccio consente di implementare controlli approfonditi per proteggere l'integrità e la disponibilità dei log e del processo di gestione dei log, distinti dagli strumenti che vi accedono. Prendi in considerazione l'utilizzo di servizi, ad esempio Amazon Athena, per eseguire query su richiesta che correlano più origini dati. Puoi anche integrare strumenti di visualizzazione, come Amazon QuickSight. Le soluzioni basate sull'intelligenza artificiale sono sempre più disponibili e possono svolgere funzioni quali la traduzione degli esiti in sintesi leggibili dall'uomo e l'interazione in linguaggio naturale. Queste soluzioni sono spesso più facilmente integrate grazie a una posizione di archiviazione di dati standardizzata per le interrogazioni.

Passaggi dell'implementazione

1. Crea gli account Log Archive e Security Tooling
    a. UtilizzandoAWS Organizations, crea gli account Log Archive e Security Tooling in un'unità organizzativa di sicurezza. Se utilizzi AWS Control Tower per gestire la tua organizzazione, gli account Log Archive e Security Tooling vengono creati automaticamente. Configura i ruoli e le autorizzazioni per l'accesso a questi account e la loro amministrazione come richiesto.

2. Configura le posizioni dei dati di sicurezza standardizzate
    a. Determina la tua strategia per la creazione di posizioni di dati di sicurezza standardizzate. È possibile ottenere questo risultato attraverso opzioni quali approcci comuni all'architettura dei data lake, prodotti di dati di terze parti o Amazon Security Lake. AWS consiglia di acquisire i dati di sicurezza dalle Regioni AWS che sono state scelte per tutti gli account, anche se non attivamente in uso.
3. Configura la pubblicazione delle origini dati nelle tue sedi standardizzate
    a. Identifica le origini dati di sicurezza e configurale per la pubblicazione nelle tue sedi standardizzate. Valuta le opzioni per esportare automaticamente i dati nel formato desiderato anziché in quelle in cui è necessario sviluppare processi ETL. Con Amazon Security Lake, puoi raccogliere dati da origini AWS supportate e sistemi integrati di terze parti.
4. Configura gli strumenti per accedere alle tue sedi standardizzate
    a. Configura strumenti come Amazon Athena, Amazon QuickSight o soluzioni di terze parti per avere l'accesso richiesto alle tue sedi standardizzate. Configura questi strumenti in modo che operino dall'account Security Tooling con accesso in lettura trasversale all'account Log Archive, se applicabile. Crea abbonati in Amazon Security Lake per fornire a questi strumenti l'accesso ai tuoi dati.

Risorse

Best practice correlate:

- SEC01-BP01 Separazione dei carichi di lavoro tramite account
- SEC07-BP04 Definizione della gestione del ciclo di vita dei dati scalabili
- SEC08-BP04 Applicazione del controllo degli accessi
- OPS08-BP02 Analizza i log relativi ai carichi di lavoro

Documenti correlati:

- AWS Whitepapers: Organizing Your AWS Environment Using Multiple Accounts
- AWS Prescriptive Guidance: AWS Security Reference Architecture (AWS SRA)
- AWS Prescriptive Guidance: Logging and monitoring guide for application owners

Esempi correlati:

- Aggregating, searching, and visualizing log data from distributed sources with Amazon Athena and Amazon QuickSight
- How to visualize Amazon Security Lake findings with Amazon QuickSight
- Generate AI powered insights for Amazon Security Lake using Amazon SageMaker Studio and Amazon Bedrock
- Identify cybersecurity anomalies in your Amazon Security Lake data using Amazon SageMaker
- Ingest, transform, and deliver events published by Amazon Security Lake to Amazon OpenSearch Service
- How to use AWS Security Hub and Amazon OpenSearch Service for SIEM

Strumenti correlati:

- Amazon Security Lake
- Amazon Security Lake Partner Integrations
- Open Cybersecurity Schema Framework (OCSF)
- Amazon Athena
- Amazon QuickSight
- Amazon Bedrock

SEC04-BP03 Correlazione e arricchimento degli avvisi di sicurezza

Un'attività inaspettata può generare più avvisi di sicurezza da origini diverse, che richiedono un'ulteriore correlazione e approfondimento per comprendere il contesto completo. Implementa la correlazione e l'approfondimento automatici degli avvisi di sicurezza per contribuire a ottenere un'identificazione e una risposta più accurate agli incidenti.

Risultato desiderato: man mano che l'attività genera diversi avvisi all'interno dei carichi di lavoro e degli ambienti, i meccanismi automatici mettono in relazione i dati e li arricchiscono con ulteriori informazioni. Questa pre-elaborazione presenta un quadro più dettagliato dell'evento, che aiuta gli investigatori a determinare la criticità dell'evento e a stabilire se si tratta di un incidente che richiede una risposta formale. Questo processo riduce il carico sui team di monitoraggio e investigazione.

Anti-pattern comuni:

- Gruppi diversi di persone esaminano i risultati e gli avvisi generati da sistemi differenti, a meno che i requisiti di separazione degli incarichi non impongano altrimenti.

- L'organizzazione convoglia tutti i dati dei risultati e degli avvisi di sicurezza in posizioni standard, ma richiede agli investigatori di eseguire correlazioni e arricchimenti manuali.
- Ti affidi esclusivamente all'intelligence dei sistemi di rilevamento delle minacce per riferire sui risultati e stabilire la criticità.

Vantaggi della definizione di questa best practice: la correlazione e l'arricchimento automatici degli avvisi contribuiscono a ridurre il carico cognitivo complessivo e la preparazione manuale dei dati richiesta agli investigatori. Questa pratica può ridurre il tempo necessario per determinare se l'evento rappresenta un incidente e avviare una risposta formale. Un contesto aggiuntivo consente inoltre di valutare con precisione la reale gravità di un evento, in quanto può essere superiore o inferiore a quanto suggerito da un avviso.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

Gli avvisi di sicurezza possono provenire da diverse fonti AWS interne, tra cui:

- Servizi come Amazon GuardDuty, AWS Security Hub, Amazon Macie, Amazon Inspector, AWS Config, AWS Identity and Access Management Access Analyzer e Network Access Analyzer
- Avvisi derivanti dall'analisi automatica dei log di servizi, infrastrutture e applicazioni AWS, ad esempio da Security Analytics per Amazon OpenSearch Service.
- Allarmi in risposta a cambiamenti nell'attività di fatturazione da fonti quali Amazon CloudWatch, Amazon EventBridge o Budget AWS.
- Fonti di terze parti come feed di intelligence sulle minacce e Soluzioni dei partner per la sicurezza di AWS Partner Network
- Contact by AWS Trust & Safety o altre origini, come i clienti o i dipendenti interni.

Nella loro forma più elementare, le segnalazioni contengono informazioni su chi (il principale o l'identità) sta facendo cosa (l'azione intrapresa) e quali sono (le risorse interessate). Per ognuna di queste origini, individua le modalità con cui puoi creare mappature tra gli identificatori per queste identità, azioni e risorse come base per eseguire la correlazione. Ciò può avvenire integrando le origini degli avvisi con uno strumento di gestione delle informazioni e degli eventi di sicurezza (SIEM) per eseguire la correlazione automatica, creando pipeline ed elaborazioni di dati proprie o una combinazione di entrambi.

Un esempio di servizio in grado di eseguire la correlazione per te è Amazon Detective. Il rilevatore inserisce continuamente avvisi da varie origini AWS e da terze parti e utilizza diverse forme di intelligenza per assemblare un grafico visivo delle loro relazioni per facilitare le indagini.

Sebbene la criticità iniziale di un avviso sia un aiuto per la definizione delle priorità, il contesto in cui l'avviso è stato generato ne determina la vera criticità. Ad esempio, Amazon GuardDuty può avvisare che un'istanza Amazon EC2 all'interno del carico di lavoro sta interrogando un nome di dominio inaspettato. GuardDuty potrebbe assegnare solo una bassa criticità a questo avviso. Tuttavia, la correlazione automatica con altre attività svolte al momento dell'allarme potrebbe rivelare che diverse centinaia di istanze EC2 sono state distribuite dalla stessa identità, con un conseguente aumento dei costi operativi complessivi. In tal caso, GuardDuty potrebbe pubblicare questo contesto di eventi correlati come un nuovo avviso di sicurezza e modificare la criticità in alta, per accelerare ulteriori azioni.

Passaggi dell'implementazione

1. Identifica le fonti di informazioni sugli avvisi di sicurezza. Scopri come gli avvisi provenienti da questi sistemi rappresentano identità, azioni e risorse per determinare dove è possibile una correlazione.
2. Stabilisci un meccanismo per acquisire avvisi da diverse origini. Considera servizi come Security Hub, EventBridge e CloudWatch per questo scopo.
3. Identifica le fonti per la correlazione e l'arricchimento dei dati. Le fonti di esempio includono CloudTrail, i log di flusso VPC, Amazon Security Lake e i log dell'infrastruttura e delle applicazioni.
4. Integra i tuoi avvisi con le tue origini di correlazione e arricchimento dei dati per creare contesti degli eventi di sicurezza più dettagliati e stabilire le criticità.
   a. Amazon Detective, strumenti SIEM o altre soluzioni di terze parti possono eseguire automaticamente un determinato livello di inserimento, correlazione e arricchimento.
   b. Puoi anche utilizzare i servizi AWS per crearne uno tuo. Ad esempio, puoi richiamare una funzione AWS Lambda per eseguire una query Amazon Athena su AWS CloudTrail o Amazon Security Lake e pubblicare i risultati su EventBridge.

Risorse

Best practice correlate:

- SEC10-BP03 Preparazione di funzionalità forensi
- OPS08-BP04 Creare avvisi fruibili

- REL06-BP03 Invio di notifiche (elaborazione e avvisi in tempo reale)

Documenti correlati:

- AWS Security Incident Response Guide

Esempi correlati:

- How to enrich AWS Security Hub findings with account metadata
- How to use AWS Security Hub and Amazon OpenSearch Service for SIEM

Strumenti correlati:

- Amazon Detective
- Amazon EventBridge
- AWS Lambda
- Amazon Athena

SEC04-BP04 Avvio della riparazione delle risorse non conformi

I controlli investigativi possono segnalare la presenza di risorse non conformi ai requisiti di configurazione. È possibile avviare interventi correttivi definiti in modo programmatico, sia manualmente che automaticamente, per riparare queste risorse e contribuire a ridurre al minimo gli impatti potenziali. Quando definisci le correzioni in modo programmatico, puoi intraprendere azioni rapide e coerenti.

Sebbene l'automazione possa migliorare le operazioni di sicurezza, è necessario implementare e gestire l'automazione con attenzione. Implementa meccanismi di supervisione e controllo appropriati per verificare che le risposte automatizzate siano efficaci, accurate e allineate alle policy organizzative e alla propensione al rischio.

Risultato desiderato: definizione degli standard di configurazione delle risorse e dei passaggi per la correzione delle risorse non conformi. Dove possibile, hai definito gli interventi correttivi in modo programmatico, affinché possano essere avviati manualmente o attraverso l'automazione. Sono disponibili sistemi di rilevamento per identificare le risorse non conformi e pubblicare avvisi in strumenti centralizzati monitorati dal personale di sicurezza. Questi strumenti supportano

l'esecuzione degli interventi programmatici, manualmente o automaticamente. Le soluzioni automatiche dispongono di meccanismi di supervisione e controllo adeguati per regolarne l'utilizzo.

Anti-pattern comuni:

- L'automazione viene implementata, ma non si riescono a testare e convalidare a fondo le azioni correttive. Ciò può comportare conseguenze indesiderate, come l'interruzione delle operazioni aziendali legittime o l'instabilità del sistema.
- I tempi e le procedure di risposta vengono migliorati grazie all'automazione, ma senza un monitoraggio adeguato e senza meccanismi che consentano l'intervento umano e il giudizio quando necessario.
- Ci si affida esclusivamente agli interventi di riparazione, senza considerarli una parte di un programma più ampio di risposta agli incidenti e di ripristino.

Vantaggi della definizione di questa best practice: le soluzioni automatiche possono rispondere alle configurazioni errate più rapidamente rispetto ai processi manuali, il che aiuta a minimizzare i potenziali impatti aziendali e a ridurre la finestra di opportunità per usi non intenzionali. Quando si definiscono le correzioni in modo programmatico, queste vengono applicate in modo coerente, riducendo il rischio di errori umani. L'automazione è inoltre in grado di gestire un volume maggiore di avvisi contemporaneamente, il che è particolarmente importante negli ambienti che operano su larga scala.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Come descritto in SEC01-BP03 Identificazione e convalida degli obiettivi di controllo, i servizi come AWS Config possono aiutarti a monitorare la configurazione delle risorse nei tuoi account per verificare che soddisfino i tuoi requisiti. Quando vengono rilevate risorse non conformi, si consiglia di configurare l'invio di avvisi a una soluzione di gestione della postura di sicurezza nel cloud (CSPM), ad esempio AWS Security Hub per facilitare la risoluzione. Queste soluzioni offrono agli investigatori della sicurezza il punto centrale per il monitoraggio dei problemi e l'adozione di misure correttive.

Mentre alcune situazioni di non conformità delle risorse sono uniche e richiedono un giudizio umano per essere risolte, altre situazioni hanno una risposta standard che si può definire in maniera programmatica. Ad esempio, una risposta standard a un gruppo di sicurezza VPC non correttamente configurato potrebbe essere la rimozione delle regole non consentite e la notifica al proprietario. Le risposte possono essere definite in funzioni AWS Lambda, documenti di AWS Systems Manager

Automation o tramite altri ambienti di codice che preferisci. Assicurati che l'ambiente sia in grado di autenticarsi ad AWS utilizzando un ruolo IAM con il minor numero di autorizzazioni necessarie per intraprendere un'azione correttiva.

Una volta definita la correzione desiderata, è possibile determinare il mezzo preferito per avviarla. AWS Config può avviare le azioni correttive per tuo conto. Se stai utilizzando Security Hub, puoi farlo tramite azioni personalizzate, che pubblicano le informazioni di ricerca su Amazon EventBridge. Una regola EventBridge può quindi avviare la correzione. È possibile configurare l'azione personalizzata in Security Hub in modo che venga eseguita automaticamente o manualmente.

Per la correzione programmatica, si consiglia di utilizzare registri e audit completi per le azioni intraprese e i relativi risultati. Rivedi e analizza questi registri per valutare l'efficacia dei processi automatizzati e identificare le aree di miglioramento. Acquisisci gli accessi Amazon CloudWatch Logs e i risultati delle correzioni come note in Security Hub.

Come punto di partenza, considera Automated Security Response su AWS, che dispone di soluzioni predefinite per risolvere le più comuni configurazioni errate della sicurezza.

Passaggi dell'implementazione

1. Analizza e assegna priorità agli avvisi.
    a. Consolida gli avvisi di sicurezza provenienti da vari servizi AWS in Security Hub per una visibilità, una definizione delle priorità e una correzione centralizzate.
2. Sviluppa soluzioni correttive.
    a. Utilizza servizi come Systems Manager e AWS Lambda per eseguire correzioni programmatiche.
3. Configura il modo in cui vengono avviate le correzioni.
    a. Utilizzando Systems Manager, definisci le azioni personalizzate che pubblicano i risultati su EventBridge. Configura queste azioni in modo che vengano avviate manualmente o automaticamente.
    b. Puoi anche utilizzare Amazon Simple Notification Service (SNS) per inviare notifiche e avvisi alle parti interessate (come il team di sicurezza o i team di risposta agli incidenti) per l'intervento manuale o l'escalation, se necessario.
4. Rivedi e analizza i log delle correzioni per verificarne l'efficacia e il miglioramento.
    a. Invia l'output del log a CloudWatch Logs. Acquisisci i risultati trovando note in Security Hub.

Risorse

Best practice correlate:

- SEC06-BP03 Riduzione della gestione manuale e dell'accesso interattivo

Documenti correlati:

- AWS Security Incident Response Guide - Detection

Esempi correlati:

- Risposta di sicurezza automatizzata su AWS
- Monitor EC2 instance key pairs using AWS Config
- Create AWS Config custom rules by using AWS CloudFormation Guard policies
- Automatically remediate unencrypted Amazon RDS DB instances and clusters

Strumenti correlati:

- AWS Systems Manager Automation
- Risposta di sicurezza automatizzata su AWS

Protezione dell’infrastruttura

Domande

- SEC 5. In che modo proteggi le risorse di rete?
- SEC 6. In che modo proteggi le risorse di calcolo?

SEC 5. In che modo proteggi le risorse di rete?

Qualsiasi carico di lavoro che abbia una qualche forma di connettività di rete, che si tratti di Internet o di una rete privata, richiede più livelli di difesa per proteggere da minacce esterne e interne basate sulla rete.

Best practice

- SEC05-BP01 Creazione di livelli di rete

- SEC05-BP02 Controllo del traffico a tutti i livelli
- SEC05-BP03 Implementazione della protezione basata sulle ispezioni
- SEC05-BP04 Automatizzazione della protezione di rete

SEC05-BP01 Creazione di livelli di rete

Segmenta la topologia di rete in diversi livelli basati su raggruppamenti logici dei componenti del carico di lavoro in base alla sensibilità dei dati e ai requisiti di accesso. Distingui tra i componenti che richiedono l'accesso in entrata da Internet, come gli endpoint Web pubblici, e quelli che necessitano solo di un accesso interno, come i database.

Risultato desiderato: i livelli della tua rete sono parte di un approccio completo di difesa stratificata alla sicurezza che completa la strategia di autenticazione e autorizzazione dell'identità dei tuoi carichi di lavoro. I livelli sono implementati in base alla sensibilità dei dati e ai requisiti di accesso, con meccanismi appropriati di flusso e controllo del traffico.

Anti-pattern comuni:

- Creazione di tutte le risorse in un VPC o una sottorete unica.
- Costruzione dei livelli di rete senza considerare i requisiti di sensibilità dei dati, il comportamento dei componenti o la loro funzionalità.
- Utilizzo di VPC e sottoreti come impostazioni predefinite per tutte le considerazioni relative al livello di rete senza considerare come i servizi gestiti da AWS influenzino la tua topologia.

Vantaggi della definizione di questa best practice: stabilire i livelli di rete è il primo passo per limitare i percorsi non necessari attraverso la rete, in particolare quelli che conducono ai sistemi e ai dati critici. In tal modo gli attori non autorizzati avranno più difficoltà ad accedere alla rete e a navigare verso altre risorse al suo interno. I livelli di rete discreti riducono l'ambito di analisi dei sistemi di ispezione, ad esempio per il rilevamento delle intrusioni o la prevenzione del malware. Di conseguenza, si riduce il potenziale di falsi positivi e il sovraccarico di elaborazione non necessario.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Quando si progetta l'architettura di un carico di lavoro, è comune separare i componenti in diversi livelli in base alle rispettive responsabilità. Ad esempio, un'applicazione Web può avere un livello di presentazione, un livello di applicazione e un livello di dati. È possibile adottare un approccio simile

quando si progetta la topologia di rete. I controlli di rete sottostanti possono contribuire a far rispettare i requisiti di accesso ai dati del carico di lavoro. Ad esempio, in un'architettura di applicazioni Web a tre livelli, puoi archiviare i file statici del livello di presentazione in Amazon S3 e servirli da una rete di distribuzione di contenuti (CDN), ad esempio Amazon CloudFront. Il livello applicativo può avere endpoint pubblici che un Application Load Balancer (ALB) serve in una sottorete Amazon VPC pubblica (simile a una zona demilitarizzata o DMZ), con servizi di back-end distribuiti in sottoreti private. Il livello dati che ospita risorse come database e file system condivisi può risiedere in sottoreti private diverse dalle risorse del livello applicativo. In corrispondenza di ciascuno di questi limiti di livello (CDN, sottorete pubblica, sottorete privata), è possibile implementare controlli che consentano solo al traffico autorizzato di attraversarli.

Analogamente alla modellazione dei livelli di rete in base allo scopo funzionale dei componenti del carico di lavoro, è necessario considerare anche la sensibilità dei dati elaborati. Utilizzando l'esempio dell'applicazione Web, mentre tutti i servizi del carico di lavoro possono risiedere all'interno del livello dell'applicazione, servizi diversi possono elaborare dati con livelli di sensibilità differenti. In questo caso, la divisione del livello dell'applicazione utilizzando più sottoreti private, diversi VPC nello stesso Account AWS o persino VPC diversi in diversi Account AWS per ogni livello di sensibilità dei dati può essere appropriata in base ai requisiti di controllo.

Un'ulteriore considerazione per i livelli di rete è la coerenza del comportamento dei componenti del carico di lavoro. Continuando l'esempio, nel livello dell'applicazione possono essere presenti servizi che accettano input dagli utenti finali o integrazioni di sistemi esterni che sono intrinsecamente più rischiosi rispetto agli input di altri servizi. A titolo di esempio, si possono citare il caricamento di file, l'esecuzione di script di codice, la scansione di e-mail e così via. La collocazione di questi servizi nel proprio livello di rete contribuisce a creare un limite di isolamento più forte attorno ad essi e può evitare che il loro comportamento unico crei falsi allarmi positivi nei sistemi di ispezione.

Come parte della progettazione, considera in che modo l'utilizzo dei servizi AWS gestiti influenza la topologia di rete. Scopri come servizi come Amazon VPC Lattice possono contribuire a semplificare l'interoperabilità dei componenti del carico di lavoro tra i livelli di rete. Durante l'utilizzo di AWS Lambda, esegui l'implementazione nelle sottoreti VPC a meno che non vi siano motivi specifici per non farlo. Determina dove si trovano gli endpoint VPC e AWS PrivateLink può semplificare l'adesione alle policy di sicurezza che limitano l'accesso ai gateway Internet.

Passaggi dell'implementazione

1. Rivedi l'architettura del carico di lavoro. Raggruppa logicamente componenti e servizi in base alle funzioni che svolgono, alla sensibilità dei dati elaborati e al loro comportamento.

2. Per i componenti che rispondono alle richieste provenienti da Internet, prendi in considerazione l'utilizzo di bilanciatori del carico o altri proxy per fornire endpoint pubblici. Esplora lo spostamento dei controlli di sicurezza utilizzando servizi gestiti, come CloudFront, Amazon API Gateway, Elastic Load Balancing e AWS Amplify per ospitare endpoint pubblici.
3. Per i componenti in esecuzione in ambienti di elaborazione, come istanze Amazon EC2, container AWS Fargate o funzioni Lambda, distribuiscili in sottoreti private in base ai tuoi gruppi sin dal primo passaggio.
4. Per i servizi AWS completamente gestiti, come Amazon DynamoDB, Amazon Kinesis o Amazon SQS, prendi in considerazione l'utilizzo di endpoint VPC come impostazione predefinita per l'accesso tramite indirizzi IP privati.

Risorse

Best practice correlate:

- REL02 Pianificazione della topologia di rete
- PERF04-BP01 In che modo la rete influisce sulle prestazioni

Video correlati:

- AWS re:Invent 2023 - AWS networking foundations

Esempi correlati:

- Esempi di VPC
- Access container applications privately on Amazon ECS by using AWS Fargate, AWS PrivateLink, and a Network Load Balancer
- Serve static content in an Amazon S3 bucket through a VPC by using Amazon CloudFront

SEC05-BP02 Controllo del traffico a tutti i livelli

All'interno dei livelli della rete, utilizza un'ulteriore segmentazione per limitare il traffico solo ai flussi necessari per ogni carico di lavoro. Innanzitutto, concentrati sul controllo del traffico tra Internet o altri sistemi esterni verso un carico di lavoro e il tuo ambiente (traffico nord-sud). Successivamente, esamina i flussi tra i diversi componenti e sistemi (traffico est-ovest).

Risultato desiderato: autorizzi solo i flussi di rete necessari ai componenti dei carichi di lavoro per comunicare tra loro, con i rispettivi client e con qualsiasi altro servizio da cui dipendono. La tua progettazione tiene conto di considerazioni come l'ingresso e l'uscita pubblici rispetto a quelli privati, la classificazione dei dati, le normative regionali e i requisiti di protocollo. Ove possibile, si preferiscono i flussi point-to-point rispetto al peering di rete come parte di un principio di progettazione dei privilegi minimi.

Anti-pattern comuni:

- Adottare un approccio alla sicurezza della rete basato sul perimetro e controllare il flusso di traffico solo al confine dei livelli di rete.
- Si presume che tutto il traffico all'interno di un livello di rete sia autenticato e autorizzato.
- I controlli si applicano al traffico in ingresso o a quello in uscita, ma non a entrambi.
- Per l'autenticazione e l'autorizzazione del traffico ci si affida esclusivamente ai componenti del carico di lavoro e ai controlli di rete.

Vantaggi dell'adozione di questa best practice: questa pratica contribuisce a ridurre il rischio di movimenti non autorizzati all'interno della rete e aggiunge un ulteriore livello di autorizzazione ai carichi di lavoro. Eseguendo il controllo del flusso di traffico, è possibile limitare la portata dell'impatto di un incidente di sicurezza e velocizzare il rilevamento e la risposta.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Se da un lato i livelli di rete aiutano a stabilire i limiti dei componenti del carico di lavoro che svolgono una funzione, un livello di sensibilità dei dati e un comportamento simili, dall'altro è possibile creare un livello di controllo del traffico molto più granulare utilizzando tecniche per segmentare ulteriormente i componenti all'interno di questi livelli, seguendo il principio del privilegio minimo. All'interno di AWS, i livelli di rete sono definiti principalmente utilizzando sottoreti in base agli intervalli di indirizzi IP all'interno di un Amazon VPC. I livelli possono anche essere definiti utilizzando diversi VPC, ad esempio per raggruppare gli ambienti di microservizi per dominio aziendale. Quando si utilizzano più VPC, mediare il routing utilizzando un AWS Transit Gateway. Sebbene ciò fornisca il controllo del traffico a livello 4 (indirizzi IP e intervalli di porte) utilizzando gruppi di sicurezza e tabelle di routing, è possibile ottenere un ulteriore controllo utilizzando servizi aggiuntivi come AWS PrivateLink, Amazon Route 53 Resolver DNS Firewall, AWS Network Firewall e AWS WAF.

Comprendi e analizza il flusso di dati e i requisiti di comunicazione dei tuoi carichi di lavoro in termini di parti che iniziano la connessione, porte, protocolli e livelli di rete. Valuta i protocolli disponibili per stabilire connessioni e trasmettere dati per selezionare quelli che soddisfano i tuoi requisiti di protezione (ad esempio, HTTPS anziché HTTP). Acquisisci questi requisiti sia ai limiti delle tue reti che all'interno di ogni livello. Una volta identificati questi requisiti, esplora le opzioni per consentire il flusso del traffico richiesto solo in ciascun punto di connessione. Un buon punto di partenza è utilizzare i gruppi di sicurezza all'interno del tuo VPC, poiché possono essere collegati a risorse che utilizzano un'interfaccia di rete elastica (ENI), come istanze Amazon EC2, attività Amazon ECS, pod Amazon EKS o database Amazon RDS. A differenza di un firewall Livello 4, un gruppo di sicurezza può avere una regola che consente il traffico da un altro gruppo di sicurezza in base al suo identificatore, riducendo al minimo gli aggiornamenti quando le risorse all'interno del gruppo cambiano nel tempo. Puoi anche filtrare il traffico utilizzando le regole in entrata e in uscita utilizzando i gruppi di sicurezza.

Quando il traffico si sposta tra i VPC, è comune utilizzare il peering VPC per il routing semplice o AWS Transit Gateway per il routing complesso. Con questi approcci, si facilitano i flussi di traffico tra l'intervallo di indirizzi IP delle reti di origine e di destinazione. Tuttavia, se il carico di lavoro richiede solo flussi di traffico tra componenti specifici in diversi VPC, considera l'utilizzo di una connessione point-to-point utilizzando AWS PrivateLink. Per fare ciò, individua quale servizio dovrebbe agire come produttore e quale dovrebbe agire come consumatore. Implementa un bilanciatore del carico compatibile per il produttore, attivalo su PrivateLink di conseguenza, quindi accetta una richiesta di connessione da parte del consumatore. Al servizio del produttore viene quindi assegnato un indirizzo IP privato dal VPC del consumatore che quest'ultimo può utilizzare per effettuare richieste successive. Questo approccio riduce la necessità di eseguire il peer-to-peer delle reti. Includi i costi per l'elaborazione dei dati e il bilanciamento del carico come parte della valutazione PrivateLink.

Mentre i gruppi di sicurezza e PrivateLink aiutano a controllare il flusso tra i componenti dei carichi di lavoro, un'altra considerazione importante è come controllare a quali domini DNS possono accedere le tue risorse (se presenti). A seconda della configurazione DHCP dei tuoi VPC, puoi prendere in considerazione due diversi servizi AWS per questo scopo. La maggior parte dei clienti utilizza il servizio Route 53 Resolver DNS predefinito (chiamato anche server Amazon DNS o AmazonProvidedDNS) disponibile per i VPC all'indirizzo +2 del suo intervallo CIDR. Con questo approccio, puoi creare regole DNS Firewall e associarle al tuo VPC per determinare quali azioni intraprendere per gli elenchi di domini che fornisci.

Se non stai utilizzando il Route 53 Resolver o se desideri integrare il Resolver con funzionalità di ispezione e controllo del flusso più approfondite oltre al filtro di dominio, prendi in considerazione l'implementazione di un AWS Network Firewall. Questo servizio ispeziona i singoli pacchetti

utilizzando regole stateless o stateful per determinare se negare o consentire il traffico. Puoi adottare un approccio simile per filtrare il traffico Web in entrata verso i tuoi endpoint pubblici utilizzando AWS WAF. Per ulteriori indicazioni su questi servizi, vedi SEC05-BP03 Implementazione della protezione basata sulle ispezioni.

Passaggi dell'implementazione

1. Identifica i flussi di dati richiesti tra i componenti dei tuoi carichi di lavoro.
2. Applica più controlli con un approccio di difesa approfondita per il traffico in entrata e in uscita, incluso l'uso di gruppi di sicurezza e tabelle di routing.
3. Usa i firewall per definire un controllo granulare sul traffico di rete in entrata, in uscita e attraverso i tuoi VPC, come Route 53 Resolver DNS Firewall, AWS Network Firewall e AWS WAF. Prendi in considerazione l'utilizzo di AWS Firewall Manager per configurare e gestire centralmente le regole del firewall in tutta l'organizzazione.

Risorse

Best practice correlate:

- REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro
- SEC09-BP02 Applicazione della crittografia dei dati in transito

Documenti correlati:

- Security best practices for your VPC
- AWS Network Optimization Tips
- Guidance for Network Security on AWS
- Secure your VPC's outbound network traffic in the Cloud AWS

Strumenti correlati:

- AWS Firewall Manager

Video correlati:

- AWS Transit Gateway reference architectures for many VPCs

- Application Acceleration and Protection with Amazon CloudFront, AWS WAF, and AWS Shield(Accelerazione e protezione delle applicazioni con Amazon CloudFront, AWS WAF e AWS Shield)
- AWS re:Inforce 2023: Firewalls and where to put them

Esempi correlati:

- Lab: CloudFront for Web Application

SEC05-BP03 Implementazione della protezione basata sulle ispezioni

Imposta i punti di ispezione del traffico tra i livelli di rete per verificare che i dati in transito corrispondano alle categorie e agli schemi previsti. Analizza i flussi di traffico, i metadati e i modelli per identificare, rilevare e rispondere agli eventi in modo più efficace.

Risultato desiderato: il traffico che attraversa i livelli di rete viene ispezionato e autorizzato. Le decisioni di autorizzazione e rifiuto si basano su regole esplicite, informazioni sulle minacce e deviazioni dai comportamenti di base. Le protezioni diventano più severe man mano che il traffico si avvicina ai dati sensibili.

Anti-pattern comuni:

- Affidarsi esclusivamente alle regole del firewall basate su porte e protocolli. Non sfruttare i sistemi intelligenti.
- Creare regole del firewall basate su specifici modelli di minaccia attuali, soggetti a modifiche.
- Ispezionare solo il traffico che transita da una sottorete privata a una pubblica o da una sottorete pubblica a Internet.
- Non avere una visione di base del traffico di rete da confrontare per individuare eventuali anomalie di comportamento.

Vantaggi dell'adozione di questa best practice: i sistemi di ispezione consentono di creare regole intelligenti, come ad esempio consentire o negare il traffico solo in presenza di determinate condizioni all'interno dei dati di traffico. Approfitta dei set di regole gestiti da AWS e dai partner, in base alle più recenti informazioni sulle minacce, in quanto il panorama delle minacce cambia nel tempo. In questo modo si riduce l'onere di mantenere le regole e di ricercare gli indicatori di compromissione, riducendo il potenziale di falsi positivi.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Ottieni un controllo granulare del traffico di rete stateful e stateless utilizzando AWS Network Firewall o altri firewall e sistemi di prevenzione delle intrusioni (IPS) su Marketplace AWS che puoi implementare dietro un GWLB. AWS Network Firewall supporta le specifiche IPS open source compatibili con Suricata per proteggere il carico di lavoro.

Sia le soluzioni AWS Network Firewall sia i fornitori che utilizzano un GWLB supportano diversi modelli di implementazione delle ispezioni in linea. Ad esempio, è possibile eseguire l'ispezione sulla base di un VPC, centralizzare in un VPC di ispezione o implementare un modello ibrido in cui il traffico est-ovest passa attraverso un VPC di ispezione e l'ingresso a Internet viene ispezionato per VPC. Un'altra considerazione è se la soluzione supporta l'unwrapping della Transport Layer Security (TLS), consentendo l'ispezione approfondita dei pacchetti per i flussi di traffico avviati in entrambe le direzioni. Per ulteriori informazioni e dettagli approfonditi su queste configurazioni, consulta la AWS Network Firewall Best Practice guide.

Se utilizzi soluzioni che eseguono ispezioni fuori banda, come l'analisi pcap dei dati dei pacchetti dalle interfacce di rete che operano in modalità promiscua, puoi configurare il traffico in mirroring nel VPC. Il traffico in mirroring viene conteggiato ai fini della larghezza di banda disponibile delle interfacce ed è soggetto agli stessi costi di trasferimento dati del traffico non in mirroring. Puoi vedere se le versioni virtuali di queste appliance sono disponibili su Marketplace AWS, in grado di supportare la distribuzione in linea dietro un GWLB.

Per i componenti che effettuano transazioni tramite protocolli basati su HTTP, proteggi la tua applicazione dalle minacce comuni con un Web Application Firewall (WAF). AWS WAF è un firewall per applicazioni Web che consente di monitorare e bloccare le richieste HTTP(S) che corrispondono alle regole configurabili prima di inviarle a Amazon API Gateway, Amazon CloudFront, AWS AppSync o Application Load Balancer. Quando valuti l'implementazione del tuo firewall per applicazioni Web, prendi in considerazione l'analisi dei pacchetti a livello applicativo (deep packet inspection), poiché alcuni richiedono la terminazione di TLS prima dell'ispezione del traffico. Per iniziare AWS WAF, puoi utilizzare le Regole gestite da AWS in combinazione con le tue o utilizzare le integrazioni dei partner esistenti.

Puoi gestire centralmente i gruppi di sicurezza AWS WAF, AWS Shield Advanced, AWS Network Firewall e Amazon VPC in tutta la tua organizzazione AWS con AWS Firewall Manager.

Passaggi dell'implementazione

1. Stabilisci se puoi applicare le regole di ispezione in modo ampio, ad esempio tramite una VPC di ispezione, o se necessiti di un approccio più granulare per VPC.
2. Per soluzioni di ispezione in linea:
   a. Se usi AWS Network Firewall, crea regole, policy firewall e il firewall stesso. Al termine della configurazione, puoi indirizzare il traffico verso l'endpoint del firewall per consentire l'ispezione.
   b. Se utilizzi un'appliance di terze parti con un Gateway Load Balancer (GWLB), implementa e configura l'appliance in una o più zone di disponibilità. Quindi, crea il tuo GWLB, il servizio endpoint, l'endpoint e configura il routing per il tuo traffico.
3. Per soluzioni di ispezione fuori banda:
   1. Attiva il mirroring del traffico VPC sulle interfacce in cui è necessario eseguire il mirroring del traffico in entrata e in uscita. È possibile utilizzare le regole Amazon EventBridge per richiamare una funzione AWS Lambda per attivare il mirroring del traffico sulle interfacce quando vengono create nuove risorse. Indirizza le sessioni di mirroring del traffico al Network Load Balancer davanti all'appliance che elabora il traffico.
4. Per soluzioni di traffico Web in entrata:
   a. Per configurare AWS WAF, inizia configurando un elenco di controllo degli accessi Web (ACL Web). L'ACL Web è una raccolta di regole con un'azione predefinita elaborata in serie (ALLOW o DENY) che definisce il modo in cui il WAF gestisce il traffico. Puoi creare regole e gruppi personalizzati o utilizzare gruppi di regole AWS gestiti nel tuo ACL Web.
   b. Una volta configurato l'ACL Web, associalo a una risorsa AWS (ad esempio Application Load Balancer, un'API REST API Gateway o una distribuzione CloudFront) per iniziare a proteggere il traffico Web.

Risorse

Documenti correlati:

- What is Traffic Mirroring?
- Implementing inline traffic inspection using third-party security appliances
- AWS Network Firewall example architectures with routing
- Centralized inspection architecture with AWS Gateway Load Balancer and AWS Transit Gateway

Esempi correlati:

- Best practices for deploying Gateway Load Balancer
- TLS inspection configuration for encrypted egress traffic and AWS Network Firewall

Strumenti correlati:

- Marketplace AWS IDS/IPS

SEC05-BP04 Automatizzazione della protezione di rete

Automatizza l'implementazione delle protezioni di rete utilizzando pratiche DevOps, come Infrastructure as code (IaC) e pipeline CI/CD. Queste pratiche possono aiutare a tenere traccia delle modifiche apportate alle protezioni di rete attraverso un sistema di controllo delle versioni, a ridurre i tempi di implementazione delle modifiche e a rilevare se le protezioni di rete si allontanano dalla configurazione desiderata.

Risultato desiderato: si definiscono le protezioni di rete con i modelli e si esegue il commit in un sistema di controllo delle versioni. Quando vengono apportate nuove modifiche, vengono avviate pipeline automatiche che ne orchestrano il test e la distribuzione. I controlli delle policy e altri test statici sono in atto per convalidare le modifiche prima dell'implementazione. Le modifiche vengono implementate in un ambiente di staging per convalidare che i controlli funzionino come previsto. Anche la distribuzione negli ambienti di produzione viene eseguita automaticamente una volta approvati i controlli.

Anti-pattern comuni:

- Affidare ai singoli team di lavoro la definizione dell'intero stack di rete, delle protezioni e delle automazioni. Non pubblicare gli aspetti standard dello stack di rete e le protezioni in modo centralizzato per consentire ai team del carico di lavoro di utilizzarli.
- Affidarsi a un team di rete centrale per definire tutti gli aspetti della rete, delle protezioni e delle automazioni. Non delegare aspetti specifici del carico di lavoro dello stack di rete e delle protezioni al team di quel carico di lavoro.
- Trovare il giusto equilibrio tra centralizzazione e delega tra un team di rete e i team del carico di lavoro, ma non applicare standard di test e implementazione coerenti nei modelli IaC e nelle pipeline CI/CD. Mancata acquisizione delle configurazioni richieste negli strumenti che controllano l'aderenza dei modelli.

Vantaggi della definizione di questa best practice: l'utilizzo di modelli per definire le protezioni di rete consente di tracciare e confrontare le modifiche nel tempo con un sistema di controllo delle versioni. L'uso dell'automazione per testare e implementare le modifiche crea standardizzazione e prevedibilità, aumentando le possibilità di una corretta implementazione e riducendo le configurazioni manuali ripetitive.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Una serie di controlli di protezione della rete descritti in SEC05-BP02 Controllo dei flussi di traffico all'interno dei livelli di rete e SEC05-BP03 Implementazione della protezione basata sulle ispezioni è dotata di sistemi di regole gestite in grado di aggiornarsi automaticamente in base alle ultime informazioni sulle minacce. Esempi di protezione degli endpoint Web includono regole AWS WAF gestite e mitigazione automatica degli attacchi DDoS a livello di applicazione AWS Shield Advanced. Utilizza i gruppi di regole AWS Network Firewall gestiti per avere aggiornamenti anche sugli elenchi di domini con scarsa reputazione e sulle firme delle minacce.

Oltre alle regole gestite, ti consigliamo di utilizzare le pratiche DevOps per automatizzare l'implementazione delle risorse di rete, delle protezioni e delle regole specificate. Puoi acquisire queste definizioni in AWS CloudFormation o in un altro strumento di Infrastructure as code (IaC) (IaC) di tua scelta, trasferirle in un sistema di controllo della versione e distribuirle utilizzando pipeline CI/CD. Usa questo approccio per ottenere i vantaggi tradizionali di DevOps per la gestione dei controlli di rete, come versioni più prevedibili, test automatici utilizzando strumenti come AWS CloudFormation Guard e rilevamento della deriva tra l'ambiente distribuito e la configurazione desiderata.

In base alle decisioni prese nell'ambito di SEC05-BP01 Creazione di livelli di rete, puoi avere un approccio di gestione centralizzato alla creazione di VPC dedicati ai flussi di ingresso, uscita e ispezione. Come descritto nella AWS Security Reference Architecture (AWSSRA), è possibile definire questi VPC in un account di infrastruttura di rete dedicato. Puoi utilizzare tecniche simili per definire centralmente i VPC utilizzati dai tuoi carichi di lavoro in altri account, i relativi gruppi di sicurezza, le distribuzioni AWS Network Firewall, le regole Route 53 Resolver e le configurazioni del firewall DNS e altre risorse di rete. Puoi condividere queste risorse con gli altri tuoi account con AWS Resource Access Manager. Con questo approccio, puoi semplificare il test e l'implementazione automatici dei controlli di rete nell'account di rete, con una sola destinazione da gestire. Puoi farlo in un modello ibrido, in cui distribuisci e condividi determinati controlli centralmente e deleghi altri controlli ai singoli team del carico di lavoro e ai rispettivi account.

Passaggi dell'implementazione

1. Stabilisci quali aspetti della rete e delle protezioni sono definiti a livello centrale e quali possono essere gestiti dai tuoi team di lavoro.
2. Crea ambienti per testare e implementare le modifiche alla tua rete e alle relative protezioni. Ad esempio, utilizza un account Network Testing e un account Network Production.
3. Determina come archivierai e manterrai i tuoi modelli in un sistema di controllo della versione. Archivia i modelli centrali in un repository distinto da quello dei carichi di lavoro, mentre i modelli dei carichi di lavoro possono essere archiviati in repository specifici per quel carico di lavoro.
4. Crea pipeline CI/CD per testare e distribuire modelli. Definisci i test per verificare che non ci siano configurazioni errate e che i modelli siano conformi agli standard aziendali.

Risorse

Best practice correlate:

- SEC01-BP06 Implementazione automatizzata dei controlli di sicurezza standard

Documenti correlati:

- AWS Security Reference Architecture - Network account

Esempi correlati:

- AWS Deployment Pipeline Reference Architecture
- NetDevSecOps to modernize AWS networking deployments
- Integrating AWS CloudFormation security tests with AWS Security Hub and AWS CodeBuild reports

Strumenti correlati:

- AWS CloudFormation
- AWS CloudFormation Guard
- cfn_nag

SEC 6. In che modo proteggi le risorse di calcolo?

Le risorse di calcolo nel carico di lavoro richiedono più livelli di difesa per contribuire alla protezione da minacce esterne ed interne. Le risorse di calcolo includono istanze EC2, container, funzioni di AWS Lambda, servizi di database, dispositivi IoT e altro.

Best practice

- SEC06-BP01 Gestione delle vulnerabilità
- SEC06-BP02 Provisioning di calcolo da immagini rafforzate
- SEC06-BP03 Riduzione della gestione manuale e dell'accesso interattivo
- SEC06-BP04 Convalida dell'integrità del software
- SEC06-BP05 Automatizzazione della protezione delle risorse di calcolo

SEC06-BP01 Gestione delle vulnerabilità

Scansiona e correggi frequentemente le vulnerabilità del codice, delle dipendenze e dell'infrastruttura per proteggere da nuove minacce.

Risultato desiderato: creare e mantenere un programma di gestione delle vulnerabilità. Esegui regolarmente scansioni e patch su risorse quali istanze Amazon EC2, container Amazon Elastic Container Service (Amazon ECS) e carichi di lavoro Amazon Elastic Kubernetes Service (Amazon EKS). Configura finestre di manutenzione per le risorse gestite da AWS, come i database Amazon Relational Database Service (Amazon RDS). Utilizza la scansione statica del codice per ispezionare il codice sorgente delle applicazioni alla ricerca di problemi comuni. Considera la possibilità di effettuare test di penetrazione (pen-test) delle applicazioni web se l'organizzazione dispone delle competenze necessarie o se può avvalersi di un'assistenza esterna.

Anti-pattern comuni:

- Assenza di un programma di gestione delle vulnerabilità.
- Esecuzione di patch di sistema senza considerare la gravità o la prevenzione del rischio.
- Utilizzo di software che ha superato la data di fine vita (EOL) prevista dal fornitore.
- Implementazione del codice in produzione prima di aver analizzato i problemi di sicurezza.

Vantaggi dell'adozione di questa best practice:

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Un programma di gestione delle vulnerabilità comprende la valutazione della sicurezza, l'identificazione dei problemi, la definizione delle priorità e l'esecuzione di operazioni di patch per risolvere i problemi. L'automazione è la chiave per la scansione continua dei carichi di lavoro alla ricerca di problemi e di esposizioni di rete non intenzionali e per l'esecuzione di interventi correttivi. L'automazione della creazione e dell'aggiornamento delle risorse fa risparmiare tempo e riduce il rischio che gli errori di configurazione creino ulteriori problemi. Un programma di gestione delle vulnerabilità ben progettato dovrebbe considerare anche la verifica delle vulnerabilità durante le fasi di sviluppo e implementazione del ciclo di vita del software. L'implementazione della gestione delle vulnerabilità durante lo sviluppo e la distribuzione aiuta a ridurre le possibilità che una vulnerabilità si diffonda nell'ambiente di produzione.

L'implementazione di un programma di gestione delle vulnerabilità richiede una buona conoscenza del Modello di responsabilità condivisa di AWS e del suo rapporto con i carichi di lavoro specifici. Secondo tale modello, AWS è responsabile della protezione dell'infrastruttura del Cloud AWS. Questa infrastruttura è composta da hardware, software, reti e strutture che eseguono i servizi Cloud AWS. La responsabilità della sicurezza nel cloud spetta a te, ad esempio per quanto riguarda i dati effettivi, la configurazione della sicurezza, le attività di gestione delle istanze Amazon EC2 e la verifica che gli oggetti Amazon S3 siano classificati e configurati correttamente. L'approccio alla gestione delle vulnerabilità può variare anche in base ai servizi utilizzati. Ad esempio, AWS gestisce l'applicazione di patch per il nostro servizio di database relazionale gestito Amazon RDS, ma tu sei responsabile dell'applicazione di patch dei database autogestiti.

AWS offre una serie di servizi per la gestione delle vulnerabilità Amazon Inspector esegue continuamente la scansione dei carichi di lavoro AWS alla ricerca di problemi software e di accessi di rete non intenzionali. AWS Systems Manager Patch Manager supporta la gestione dell'applicazione di patch sulle istanze Amazon EC2. Amazon Inspector e Systems Manager possono essere visualizzati in AWS Security Hub, un servizio di gestione della postura di sicurezza del cloud che aiuta ad automatizzare i controlli di sicurezza AWS e a centralizzare gli avvisi di sicurezza.

Amazon CodeGuru può aiutare a identificare potenziali problemi nelle applicazioni Java e Python utilizzando l'analisi statica del codice.

Passaggi dell'implementazione

- Configurare Amazon Inspector: Amazon Inspector rileva automaticamente le istanze Amazon EC2 appena lanciate, le funzioni Lambda e le immagini di container idonee inviate ad Amazon ECR e

le analizza immediatamente alla ricerca di problemi di software, potenziali difetti ed esposizione di rete non intenzionale.

- Eseguire la scansione del codice sorgente: esegui la scansione delle librerie e delle dipendenze alla ricerca di problemi e difetti. Amazon CodeGuru può scansionare e fornire consigli per risolvere i problemi di sicurezza più comuni per le applicazioni Java e Python. OWASP Foundation pubblica un elenco di strumenti per l'analisi del codice sorgente (noti anche come strumenti SAST).
- Implementare un processo che consenta di eseguire la scansione dell'ambiente e di applicarvi le patch, nonché di eseguire la scansione come parte di un processo di compilazione di una pipeline CI/CD: implementa un processo per la scansione e l'applicazione di patch per i problemi delle dipendenze e dei sistemi operativi per proteggerti dalle nuove minacce. Tale processo deve essere eseguito regolarmente. La gestione delle vulnerabilità del software è essenziale per capire dove è necessario applicare le patch o risolvere i problemi del software. Stabilisci le priorità per la correzione di potenziali problemi di sicurezza incorporando le valutazioni di vulnerabilità nelle fasi iniziali della pipeline di integrazione continua/consegna continua (CI/ CD). L'approccio può variare in base ai servizi AWS utilizzati. Per verificare la presenza di potenziali problemi nel software in esecuzione nelle istanze Amazon EC2, aggiungi Amazon Inspector alla pipeline per avvisare l'utente e interrompere il processo di creazione se vengono rilevati problemi o potenziali difetti. Amazon Inspector monitora le risorse in modo continuo. Puoi anche utilizzare i prodotti open source come OWASP Dependency-Check, Snyk, OpenVAS, i sistemi di gestione dei pacchetti e gli strumenti AWS Partner per la gestione delle vulnerabilità.
- Utilizza AWS Systems Manager: sei responsabile della gestione delle patch per le risorse AWS, incluse le istanze Amazon Elastic Compute Cloud (Amazon EC2), le Amazon Machine Image (AMI) e le altre risorse di calcolo. AWS Systems Manager Patch Manager automatizza il processo di patch delle istanze gestite con aggiornamenti di sicurezza e di altro tipo. Patch Manager può essere utilizzato per applicare le patch alle istanze Amazon EC2 sia per i sistemi operativi che per le applicazioni, inclusi applicazioni Microsoft, service pack di Windows e aggiornamenti di versione minori per le istanze basate su Linux. Oltre a Amazon EC2, Patch Manager può essere utilizzato anche per applicare patch ai server on-premise.

  Per avere un elenco dei sistemi operativi supportati, consulta Sistemi operativi supportati nella Guida per l'utente di Systems Manager. Puoi eseguire la scansione delle istanze per visualizzare solo un report delle patch mancanti oppure puoi eseguire la scansione e installare automaticamente tutte le patch mancanti.
- Utilizzare AWS Security Hub: Security Hub offre una visione completa dello stato di sicurezza in AWS. Raccoglie i dati di sicurezza su più servizi AWS e fornisce tali risultati in un formato standardizzato, consentendo di dare priorità ai risultati della sicurezza tra i servizi AWS.

- Utilizzare AWS CloudFormation: AWS CloudFormation è un servizio Infrastruttura come codice (IaC) che può essere d'aiuto nella gestione delle vulnerabilità, automatizzando l'implementazione delle risorse e standardizzando l'architettura delle risorse tra più account e ambienti.

Risorse

Documenti correlati:

- AWS Systems Manager
- Security Overview of AWS Lambda (Panoramica sulla sicurezza di AWS Lambda)
- Amazon CodeGuru
- Improved, Automated Vulnerability Management for Cloud Workloads with a New Amazon Inspector (Gestione delle vulnerabilità migliorata e automatizzata per i carichi di lavoro cloud con un nuovo Amazon Inspector)
- Automate vulnerability management and remediation in AWS using Amazon Inspector and AWS Systems Manager – Part 1 (Automatizzare la gestione delle vulnerabilità e la bonifica in AWS utilizzando Amazon Inspector e AWS Systems Manager - Parte 1)

Video correlati:

- Securing Serverless and Container Services
- Security best practices for the Amazon EC2 instance metadata service (Best practice di sicurezza per il servizio di metadati dell'istanza Amazon EC2)

SEC06-BP02 Provisioning di calcolo da immagini rafforzate

Riduci le opportunità di accesso involontario agli ambienti di runtime implementandoli da immagini rafforzate. Acquisisci dipendenze di runtime, come immagini di container e librerie di applicazioni, solo da registri affidabili e verifica le loro firme. Crea i tuoi registri privati per archiviare immagini e librerie attendibili da utilizzare nei tuoi processi di compilazione e implementazione.

Risultato desiderato: le risorse di calcolo vengono fornite da immagini di base rafforzate. Le dipendenze esterne, ad esempio le immagini dei container e le librerie di applicazioni, vengono recuperate solo da registri attendibili e ne vengono verificate le firme. Queste sono archiviate in registri privati a cui i processi di compilazione e implementazione possono fare riferimento. Scansiona e aggiorna regolarmente immagini e dipendenze per proteggerti da eventuali vulnerabilità scoperte di recente.

Anti-pattern comuni:

- Acquisire immagini e librerie da registri attendibili, ma senza verificarne la firma o eseguire scansioni delle vulnerabilità prima di metterle in uso.
- Rafforzare le immagini, ma non testarle regolarmente per individuare nuove vulnerabilità o aggiornarle alla versione più recente.
- Installare o non rimuovere pacchetti software non necessari durante il ciclo di vita previsto dell'immagine.
- Affidarsi esclusivamente alle patch per mantenere aggiornate le risorse di calcolo di produzione. La sola applicazione di patch può comunque far sì che nel tempo le risorse di calcolo si allontanino dallo standard protetto. L'applicazione delle patch può inoltre non riuscire a rimuovere le minacce informatiche che potrebbero essere state installate da un attore pericoloso durante un evento di sicurezza.

Vantaggi derivanti dall'adozione di questa best practice: il rafforzamento delle immagini aiuta a ridurre il numero di percorsi disponibili nell'ambiente di runtime che possono consentire l'accesso involontario a utenti o servizi non autorizzati. Inoltre, può ridurre l'ambito dell'impatto in caso di accesso involontario.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per rendere i sistemi più resistenti, è necessario partire dalle versioni più recenti dei sistemi operativi, delle immagini dei container e delle librerie delle applicazioni. Applica le patch ai problemi noti. Riduci al minimo il sistema rimuovendo le applicazioni, i servizi, i driver dei dispositivi, gli utenti predefiniti e altre credenziali non necessarie. Adotta qualsiasi altra azione necessaria, come la disabilitazione delle porte, per creare un ambiente che disponga solo delle risorse e delle capacità necessarie per i carichi di lavoro. Da questa linea di base è possibile installare software, agenti o altri processi necessari per scopi quali il monitoraggio del carico di lavoro o la gestione delle vulnerabilità.

È possibile ridurre l'onere del rafforzamento dei sistemi utilizzando le linee guida fornite da origini attendibili, come il Center for Internet Security (CIS) e le Defense Information Systems Agency (DISA) Security Technical Implementation Guides (STIGs). Ti consigliamo di iniziare con una Amazon Machine Image (AMI) pubblicata da AWS o un partner APN e di utilizzare AWS EC2 Image Builder per automatizzare la configurazione in base a una combinazione appropriata di controlli CIS e STIG.

Sebbene siano disponibili immagini e ricette EC2 Image Builder rafforzate che applicano i consigli CIS o DISA STIG, è possibile che la loro configurazione impedisca il corretto funzionamento del software. In questa situazione, è possibile partire da un'immagine di base non temprata, installare il software e quindi applicare in modo incrementale i controlli CIS per verificarne l'impatto. Per qualsiasi controllo CIS che impedisca l'esecuzione del software, verifica se è possibile implementare le raccomandazioni di rafforzamento granulare in una DISA. Tieni traccia dei diversi controlli CIS e delle configurazioni DISA STIG che puoi applicare con successo. Usa tutto questo per definire le ricette di rafforzamento dell'immagine in EC2 Image Builder.

Per i carichi di lavoro containerizzati, sono disponibili le immagini rafforzate di Docker nel repository pubblico Amazon Elastic Container Registry. È possibile utilizzare EC2 Image Builder per rafforzare le immagini dei container insieme alle AMI.

In modo simile ai sistemi operativi e alle immagini dei container, è possibile ottenere pacchetti di codice (o librerie) da repository pubblici, attraverso strumenti come pip, npm, Maven e NuGet. È consigliabile gestire i pacchetti di codice integrando repository privati, come quelli all'interno di AWS CodeArtifact, con repository pubblici affidabili. Questa integrazione può gestire il recupero, l'archiviazione e l'aggiornamento dei pacchetti per l'utente. I processi di creazione dell'applicazione possono quindi ottenere e testare l'ultima versione di questi pacchetti insieme all'applicazione, utilizzando tecniche come la Software Composition Analysis (SCA), lo Static Application Security Testing (SAST) e il Dynamic Application Security Testing (DAST).

Per i carichi di lavoro serverless che utilizzano AWS Lambda, semplifica la gestione delle dipendenze dei pacchetti utilizzando i livelli Lambda. Usa i livelli Lambda per configurare un set di dipendenze standard condivise tra diverse funzioni in un archivio autonomo. È possibile creare e gestire i livelli tramite il relativo processo di costruzione, fornendo un modo centralizzato per mantenere aggiornate le funzioni.

Passaggi dell'implementazione

- Rafforzamento del sistema operativo. Utilizza immagini di base provenienti da fonti affidabili come base per costruire AMI protette. Utilizzale EC2 Image Builder per personalizzare il software installato sulle tue immagini.
- Rafforzamento delle risorse containerizzate. Configura le risorse containerizzate per il rispetto delle best practice in materia di sicurezza. Quando utilizzi i container, implementa la scansione delle immagini ECR nella pipeline di costruzione e su base regolare nel repository di immagini per cercare le CVE nei container.

- Quando utilizzi l'implementazione serverless con AWS Lambda, utilizza i livelli Lambda per separare il codice funzione dell'applicazione e le librerie dipendenti condivise. Configura la firma del codice per Lambda per assicurarti che nelle tue funzioni Lambda venga eseguito solo codice affidabile.

Risorse

Best practice correlate:

- OPS05-BP05 Esecuzione della gestione delle patch

Video correlati:

- Deep dive into AWS Lambda security

Esempi correlati:

- Quickly build STIG-compliant AMI using EC2 Image Builder
- Building better container images
- Using Lambda layers to simplify your development process
- Develop & Deploy AWS Lambda Layers using Serverless Framework
- Building end-to-end AWS DevSecOps CI/CD pipeline with open source SCA, SAST and DAST tools

SEC06-BP03 Riduzione della gestione manuale e dell'accesso interattivo

Utilizza l'automazione per eseguire attività di implementazione, configurazione, manutenzione e investigazione, laddove possibile. Quando l'automazione non è disponibile, considera l'accesso manuale alle risorse di calcolo in caso di procedure di emergenza o in ambienti sicuri (sandbox).

Risultato desiderato: gli script programmatici e i documenti di automazione (runbook) acquisiscono le azioni autorizzate sulle risorse di calcolo. Questi runbook vengono avviati automaticamente, attraverso i sistemi di rilevamento delle modifiche, o manualmente, quando è necessario il giudizio umano. L'accesso diretto alle risorse di calcolo è disponibile solo in situazioni di emergenza, quando l'automazione non è disponibile. Tutte le attività manuali vengono registrate e inserite in un processo di revisione per migliorare continuamente le capacità di automazione.

Anti-pattern comuni:

- Accesso interattivo alle istanze Amazon EC2 con protocolli come SSH o RDP.
- Gestione degli accessi dei singoli utenti come `/etc/passwd` o degli utenti locali di Windows.
- Condivisione di una password o chiave privata per accedere a un'istanza tra più utenti.
- Installazione del software e creazione o aggiornamento manuali dei file di configurazione.
- Aggiornamento o applicazione di patch manuale al software.
- Accesso a un'istanza per risolvere i problemi.

Vantaggi della definizione di questa best practice: eseguire azioni con l'automazione aiuta a ridurre il rischio operativo di modifiche non volute e di configurazioni errate. Abolire l'uso di Secure Shell (SSH) e Remote Desktop Protocol (RDP) per l'accesso interattivo significa ridurre la portata dell'accesso alle risorse di calcolo. In tal modo si elimina un percorso comune per le azioni non autorizzate. Acquisire le attività di gestione delle risorse di calcolo in documenti di automazione e script di programmazione significa definire e verificare l'intero ambito delle attività autorizzate a un livello di dettaglio granulare.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

L'accesso a un'istanza è un approccio classico all'amministrazione del sistema. Dopo aver installato il sistema operativo del server, gli utenti in genere accedono manualmente per configurare il sistema e installare il software desiderato. Nel corso del ciclo di vita del server, gli utenti possono accedere per eseguire aggiornamenti del software, applicare patch, modificare le configurazioni e risolvere i problemi.

L'accesso manuale comporta tuttavia una serie di rischi. Richiede un server in grado di ascoltare le richieste, come un servizio SSH o RDP, in grado di fornire un potenziale percorso di accesso non autorizzato. Inoltre, aumenta il rischio di errore umano associato all'esecuzione di operazioni manuali. Le conseguenze possono essere incidenti sul carico di lavoro, danneggiamento o distruzione dei dati o altri problemi di sicurezza. L'accesso umano richiede anche protezioni contro la condivisione delle credenziali, creando ulteriori costi di gestione.

Per mitigare questi rischi, puoi implementare una soluzione di accesso remoto basata su agenti, ad esempio AWS Systems Manager. AWS Systems Manager Agent (SSM Agent) avvia un canale crittografato, pertanto non si avvale dell'ascolto di richieste esterne. Prendi in considerazione la possibilità di configurare SSM Agent per stabilire questo canale su un endpoint VPC.

Systems Manager consente un controllo granulare delle modalità di interazione con le istanze gestite. In questo modo è possibile definire le automazioni da eseguire, chi può eseguirle e quando. Systems Manager può applicare patch, installare software e apportare modifiche alla configurazione senza accedere in modo interattivo all'istanza. Systems Manager può inoltre fornire l'accesso a una shell remota e registrare ogni comando invocato, e il relativo output, durante la sessione nei log e in Amazon S3. AWS CloudTrail registra le invocazioni delle API Systems Manager per l'ispezione.

Passaggi dell'implementazione

1. Installa AWS Systems Manager Agent (SSM Agent) sulle tue istanze Amazon EC2. Verifica se SSM Agent è incluso e avviato automaticamente come parte della configurazione AMI di base.
2. Verifica che i ruoli IAM associati ai profili delle tue istanze EC2 includano la policy gestita `AmazonSSMManagedInstanceCore` di IAM.
3. Disabilita SSH, RDP e altri servizi di accesso remoto in esecuzione sulle tue istanze. Puoi farlo eseguendo script configurati nella sezione dei dati utente dei tuoi modelli di avvio o creando AMI personalizzate con strumenti come EC2 Image Builder.
4. Verifica che le regole di ingresso del gruppo di sicurezza applicabili alle tue istanze EC2 non consentano l'accesso sulla porta 22/tcp (SSH) o sulla porta 3389/tcp (RDP). Implementa il rilevamento e l'invio di avvisi su gruppi di sicurezza non configurati correttamente utilizzando servizi come AWS Config.
5. Definisci automazioni, runbook ed esegui comandi appropriati in Systems Manager. Utilizza le policy IAM per definire chi può eseguire queste azioni e le condizioni in base alle quali sono consentite. Testa accuratamente queste automazioni in un ambiente non di produzione. Richiama queste automazioni quando necessario, invece di accedere in modo interattivo all'istanza.
6. Utilizza AWS Systems Manager Session Manager per fornire l'accesso interattivo alle istanze quando necessario. Attiva la registrazione delle attività della sessione per mantenere un audit trail in Amazon CloudWatch Logs o Amazon S3.

Risorse

Best practice correlate:

- REL08-BP04 Esecuzione dell'implementazione utilizzando un'infrastruttura immutabile

Esempi correlati:

- Sostituzione dell'accesso SSH per ridurre il sovraccarico di gestione e sicurezza con AWS Systems Manager

Strumenti correlati:

- AWS Systems Manager

Video correlati:

- Controlling User Session Access to Instances in AWS Systems Manager Session Manager

SEC06-BP04 Convalida dell'integrità del software

Utilizza la verifica crittografica per convalidare l'integrità degli artefatti software (comprese le immagini) utilizzati dal tuo carico di lavoro. La firma crittografica del software è una garanzia contro le modifiche non autorizzate eseguite negli ambienti di calcolo.

Risultato desiderato: tutti gli artefatti sono ottenuti da fonti attendibili. I certificati del sito Web del fornitore sono convalidati. Gli artefatti scaricati vengono verificati crittograficamente tramite le relative firme. Il tuo software è firmato e verificato crittograficamente dai tuoi ambienti informatici.

Anti-pattern comuni:

- Affidarsi a siti Web di fornitori attendibili per ottenere artefatti software, ma ignorare gli avvisi di scadenza dei certificati. Procedere al download senza confermare la validità dei certificati.
- Convalidare i certificati dei siti Web dei fornitori, ma non verificare crittograficamente gli artefatti scaricati da questi siti Web.
- Affidarsi esclusivamente a digest o hash per convalidare l'integrità del software. Gli hash stabiliscono che gli artefatti non sono stati modificati rispetto alla versione originale, ma non ne convalidano l'origine.
- Non firmare il software, il codice o le librerie di proprietà, anche se utilizzati solo per le proprie implementazioni.

Vantaggi dell'adozione di questa best practice: la convalida dell'integrità degli artefatti da cui dipende il carico di lavoro aiuta a prevenire l'ingresso di malware negli ambienti di calcolo. La firma del software aiuta a proteggerti dall'esecuzione non autorizzata nei tuoi ambienti di calcolo. Proteggi la catena di fornitura del software firmando e verificando il codice.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Le immagini del sistema operativo, le immagini dei container e gli artefatti del codice sono spesso distribuiti con controlli di integrità disponibili, ad esempio attraverso un digest o un hash. Questi permettono ai client di verificare l'integrità calcolando il proprio hash del payload e verificando che sia uguale a quello pubblicato. Sebbene questi controlli aiutino a verificare che il payload non sia stato alterato, non ne convalidano la provenienza dalla sua provenienza originaria. La verifica della provenienza richiede un certificato rilasciato da un'autorità attendibile per firmare digitalmente l'artefatto.

Se utilizzi un software o artefatti scaricati nel tuo carico di lavoro, controlla se il fornitore offre una chiave pubblica per la verifica della firma digitale. Ecco alcuni esempi di come AWS fornisce una chiave pubblica e le istruzioni di verifica per il software che pubblichiamo:

- EC2 Image Builder: Verify the signature of the AWSTOE installation download
- AWS Systems Manager: Verifying the signature of SSM Agent
- Amazon CloudWatch: Verifying the signature of the CloudWatch agent package

Incorpora la verifica della firma digitale nei processi utilizzati per ottenere e rafforzare le immagini, come discusso in SEC06-BP02 Provisioning di calcolo da immagini rafforzate.

Puoi utilizzare AWS Signer per gestire la verifica delle firme e il ciclo di vita della firma del codice per il tuo software e i tuoi artefatti. AWS Lambda e Amazon Elastic Container Registry forniscono entrambi integrazioni con Signer per verificare le firme del codice e delle immagini. Utilizzando gli esempi nella sezione Risorse, puoi incorporare Signer nelle tue pipeline di integrazione e distribuzione continua (CI/CD) per automatizzare la verifica delle firme e la firma del tuo codice e delle tue immagini.

Risorse

Documenti correlati:

- Cryptographic Signing for Containers
- Best Practices to help secure your container image build pipeline by using AWS Signer
- Announcing Container Image Signing with AWS Signer and Amazon EKS
- Configuring code signing for AWS Lambda

- Best practices and advanced patterns for Lambda code signing
- Code signing using AWS Certificate Manager Private CA and AWS Key Management Service asymmetric keys

Esempi correlati:

- Automate Lambda code signing with Amazon CodeCatalyst and AWS Signer
- Signing and Validating OCI Artifacts with AWS Signer

Strumenti correlati:

- AWS Lambda
- AWS Signer
- AWS Certificate Manager
- AWS Key Management Service
- AWS CodeArtifact

SEC06-BP05 Automatizzazione della protezione delle risorse di calcolo

Automatizza le operazioni di protezione delle risorse di calcolo per ridurre la necessità di intervento umano. Usa la scansione automatica per rilevare potenziali problemi all'interno delle tue risorse di calcolo e rimedia con risposte programmatiche automatiche o operazioni di gestione del parco. Incorpora l'automazione nei tuoi processi CI/CD per implementare carichi di lavoro affidabili con dipendenze aggiornate.

Risultato desiderato: i sistemi automatici eseguono tutte le scansioni e le patch delle risorse di calcolo. Si utilizza la verifica automatica per controllare che le immagini e le dipendenze del software provengano da fonti affidabili e non siano state manomesse. I carichi di lavoro vengono controllati automaticamente per verificare la presenza di dipendenze aggiornate e vengono firmati per stabilire l'affidabilità negli ambienti di calcolo AWS. Le correzioni automatiche vengono avviate al rilevamento di risorse non conformi.

Anti-pattern comuni:

- Adottare la pratica dell'infrastruttura immutabile, senza però disporre di una soluzione di patch di emergenza o di sostituzione dei sistemi di produzione.

- Utilizzare l'automazione per correggere le risorse non correttamente configurate, ma non avere un meccanismo di annullamento manuale. Possono verificarsi situazioni in cui è necessario modificare i requisiti e sospendere le automazioni fino a quando non si modificano.

Vantaggi derivanti dall'adozione di questa best practice: l'automazione può ridurre il rischio di accesso alle risorse di calcolo non autorizzato e del loro utilizzo. Contribuisce a evitare che le configurazioni errate si diffondano negli ambienti di produzione e a rilevare e correggere tali configurazioni nel caso in cui si verifichino. L'automazione aiuta anche a rilevare l'accesso non autorizzato delle risorse di calcolo e il loro utilizzo, riducendo i tempi di risposta. In questo modo è possibile ridurre la portata complessiva dell'impatto del problema.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

È possibile applicare le automazioni descritte nelle pratiche del principio della sicurezza per proteggere le risorse di calcolo. SEC06-BP01 Gestione delle vulnerabilità descrive come utilizzare Amazon Inspector sia nelle pipeline CI/CD sia per la scansione continua degli ambienti di runtime alla ricerca di CVE (Common Vulnerabilities and Exposures). Puoi utilizzare AWS Systems Manager per applicare le patch o per eseguire una nuova implementazione da immagini nuove attraverso runbook automatizzati per mantenere il parco computer aggiornato con il software e le librerie più recenti. Utilizza queste tecniche per ridurre la necessità di processi manuali e l'accesso interattivo alle tue risorse di elaborazione. Vedi SEC06-BP03 Riduzione della gestione manuale e dell'accesso interattivo per saperne di più.

L'automazione svolge anche un ruolo nell'implementazione di carichi di lavoro affidabili, descritti in SEC06-BP02 Provisioning di calcolo da immagini rafforzate e SEC06-BP04 Convalida dell'integrità del software. Puoi utilizzare servizi come EC2 Image Builder, AWS Signer, AWS CodeArtifact e Amazon Elastic Container Registry (ECR) per scaricare, verificare, creare e archiviare immagini consolidate e approvate e dipendenze di codice. Oltre a Inspector, ognuno di questi può svolgere un ruolo nel processo CI/CD, in modo che il carico di lavoro arrivi in produzione solo quando è confermato che le sue dipendenze sono aggiornate e provengono da fonti affidabili. Il carico di lavoro è inoltre firmato in modo che gli ambienti di calcolo AWS, come AWS Lambda e Amazon Elastic Kubernetes Service (EKS) possano verificare che non sia stato manomesso prima di consentirne l'esecuzione.

Oltre a questi controlli preventivi, è possibile utilizzare l'automazione nei controlli investigativi anche per le risorse di calcolo. Ad esempio, AWS Security Hub offre lo standard NIST 800-53 Rev. 5 che

include controlli come [EC2.8] istanze EC2 che dovrebbero utilizzare Instance Metadata Service Version 2 (IMDSv2). IMDSv2 utilizza le tecniche di autenticazione della sessione, il blocco delle richieste che contengono un'intestazione X-Forwarded-For HTTP e un TTL di rete pari a 1 per bloccare il traffico proveniente da fonti esterne per recuperare informazioni sull'istanza EC2. Questo controllo in Security Hub può rilevare quando le istanze EC2 utilizzano IMDSv1 e avviare una correzione automatica. Scopri di più sul rilevamento automatico e sulle correzioni in SEC04-BP04 Avvio della riparazione delle risorse non conformi.

Passaggi dell'implementazione

1. Automatizza la creazione di AMI sicure, conformi e rafforzate con EC2 Image Builder È possibile produrre immagini che incorporano i controlli dei Benchmark del Center for Internet Security (CIS) o gli standard della Security Technical Implementation Guide (STIG) dalle immagini di base di AWS e dei partner APN.
2. Automatizzazione della gestione delle configurazioni. Applica e convalida automaticamente le configurazioni sicure nelle risorse di calcolo utilizzando un servizio o uno strumento di gestione della configurazione.
    a. Gestione automatizzata della configurazione tramite AWS Config
    b. Gestione automatizzata della sicurezza e della conformità utilizzando AWS Security Hub
3. Automatizza l'applicazione di patch o la sostituzione delle istanze Amazon Elastic Compute Cloud (Amazon EC2). AWS Systems Manager Patch Manager automatizza il processo di patch delle istanze gestite con aggiornamenti di sicurezza e di altro tipo. Puoi utilizzare il gestore patch per applicare patch sia per i sistemi operativi sia per le applicazioni.
    a. AWS Systems Manager Patch Manager
4. Automatizza la scansione delle risorse di calcolo alla ricerca di CVE (Common Vulnerabilities and Exposures) e integra le soluzioni di scansione della sicurezza nella tua pipeline di compilazione.
    a. Amazon Inspector
    b. scansione delle immagini ECR
5. Prendi in considerazione Amazon GuardDuty per il rilevamento automatico di malware e minacce per proteggere le risorse di calcolo. GuardDuty può anche identificare potenziali problemi quando una funzione AWS Lambda viene richiamata nel tuo ambiente AWS.
    a. Amazon GuardDuty
6. Prendi in considerazione le soluzioni dei partner AWS. I partner AWS offrono prodotti leader nel settore che sono equivalenti, identici o si integrano ai controlli esistenti negli ambienti on-premise.

Questi prodotti integrano i servizi AWS esistenti per permettere di implementare un'architettura di sicurezza completa e un'esperienza più fluida nel cloud e negli ambienti on-premise.

a. Sicurezza dell'infrastruttura

Risorse

Best practice correlate:

- SEC01-BP06 Implementazione automatizzata dei controlli di sicurezza standard

Documenti correlati:

- Get the full benefits of IMDSv2 and disable IMDSv1 across your AWS infrastructure

Video correlati:

- Security best practices for the Amazon EC2 instance metadata service (Best practice di sicurezza per il servizio di metadati dell'istanza Amazon EC2)

Protezione dei dati

Domande

- SEC 7. In che modo classifichi i dati?
- SEC 8. In che modo proteggi i dati inattivi?
- SEC 9. In che modo proteggi i dati in transito?

SEC 7. In che modo classifichi i dati?

La classificazione fornisce un modo per categorizzare i dati in base ai livelli di criticità e sensibilità, in modo da aiutarti a determinare i controlli di protezione e conservazione appropriati.

Best practice

- SEC07-BP01 Comprendere lo schema di classificazione dei dati
- SEC07-BP02 Applicazione di controlli di protezione dei dati in base alla loro sensibilità
- SEC07-BP03 Automazione dell'identificazione e della classificazione

- SEC07-BP04 Definizione della gestione del ciclo di vita dei dati scalabili

SEC07-BP01 Comprendere lo schema di classificazione dei dati

Comprendi la classificazione dei dati che il tuo carico di lavoro sta elaborando, i requisiti di gestione, i processi aziendali associati, dove sono archiviati i dati e chi è il proprietario dei dati. Lo schema di classificazione e gestione dei dati deve tenere conto dei requisiti legali e di conformità applicabili del carico di lavoro e dei controlli dei dati necessari. Comprendere i dati è il primo passo nel percorso della classificazione dei dati.

Risultato desiderato: i tipi di dati presenti nel carico di lavoro sono ben compresi e documentati. Sono in atto controlli appropriati per proteggere i dati sensibili in base alla loro classificazione. Questi controlli regolano considerazioni quali chi è autorizzato ad accedere ai dati e per quale scopo, dove vengono archiviati i dati, qual è la policy di crittografia per tali dati e come vengono gestite le chiavi di crittografia, il ciclo di vita dei dati e i requisiti di conservazione, i processi di distruzione appropriati, i processi di backup e ripristino in atto e la verifica degli accessi.

Anti-pattern comuni:

- Non disporre di una policy formale di classificazione dei dati per definire i livelli di sensibilità dei dati e i relativi requisiti di gestione.
- Non avere una corretta consapevolezza dei livelli di sensibilità dei dati all'interno del carico di lavoro e non catturare queste informazioni nella documentazione dell'architettura e delle operazioni.
- Non riuscire ad applicare i controlli appropriati sui dati in base alla loro sensibilità e ai requisiti, come indicato nella relativa policy di classificazione e trattamento.
- Non riuscire a fornire un feedback sui requisiti di classificazione e trattamento dei dati ai proprietari delle policy.

Vantaggi della definizione di questa best practice: questa pratica elimina le ambiguità sulla corretta gestione dei dati all'interno del carico di lavoro. L'applicazione di una policy formale che definisca i livelli di sensibilità dei dati nella propria organizzazione e le relative protezioni richieste, può aiutare a rispettare le normative legali e altre attestazioni e certificazioni di sicurezza informatica. I proprietari dei carichi di lavoro possono avere la certezza di sapere dove sono archiviati i dati sensibili e quali controlli di protezione sono in atto. La loro acquisizione nella documentazione aiuta i nuovi membri del team a comprenderli meglio e a gestire i controlli nelle prime fasi del loro mandato. Queste

pratiche possono anche aiutare a ridurre i costi dimensionando correttamente i controlli per ogni tipo di dati.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Quando si progetta un carico di lavoro, si può pensare a come proteggere i dati sensibili in modo intuitivo. Ad esempio, in un'applicazione multi-tenant, è intuitivo considerare i dati di ogni tenant come sensibili e mettere in atto protezioni in modo che un tenant non possa accedere ai dati di un altro tenant. Allo stesso modo, è possibile progettare intuitivamente i controlli di accesso in modo che solo gli amministratori possano modificare i dati, e che gli altri utenti abbiano solo accesso a livello di lettura o non abbiano alcun accesso.

La definizione e l'acquisizione di questi livelli di sensibilità dei dati nelle policy, insieme ai relativi requisiti di protezione dei dati, consente di identificare formalmente quali dati risiedono nel carico di lavoro. È quindi possibile determinare se sono stati predisposti i controlli giusti, se i controlli possono essere verificati e quali sono le risposte appropriate in caso di gestione errata dei dati.

Per aiutarti a classificare dove sono presenti dati sensibili all'interno del carico di lavoro, valuta la possibilità di utilizzare i tag delle risorse laddove disponibili. Ad esempio, puoi applicare un tag con una chiave di `classificazione` e un valore di tag di `PHI` per informazioni sullo stato protette (PHI) e un altro tag con una chiave di tag di `sensibilità` e un valore di tag `alto`. Servizi come AWS Config possono quindi essere utilizzati per monitorare le modifiche di queste risorse e avvisare in caso di modifica in modo da renderle non conformi ai requisiti di protezione dell'utente (ad esempio la modifica delle impostazioni di crittografia). Puoi acquisire la definizione standard delle chiavi dei tag e dei valori accettabili utilizzando le policy dei tag, una funzionalità di AWS Organizations. Non è consigliabile che la chiave o il valore dei tag contenga dati privati o sensibili.

Passaggi dell'implementazione

1. Comprendi lo schema di classificazione dei dati e i requisiti di protezione della tua organizzazione.
2. Identifica i tipi di dati sensibili elaborati dai tuoi carichi di lavoro.
3. Verifica che i dati sensibili siano archiviati e protetti all'interno del tuo carico di lavoro in base alla tua policy. Utilizza tecniche come i test automatici per verificare l'efficacia dei tuoi controlli.
4. Prendi in considerazione l'utilizzo di tag a livello di risorse e dati, laddove disponibili, per etichettare i dati con il relativo livello di sensibilità e altri metadati operativi che possono aiutare nel monitoraggio e nella risposta agli incidenti.

a. Le policy dei tag AWS Organizations possono essere utilizzate per applicare gli standard di etichettatura.

Risorse

Best practice correlate:

- SUS04-BP01 Implementazione di una policy di classificazione dei dati

Documenti correlati:

- Whitepaper sulla classificazione dei dati
- Best Practices for Tagging AWS Resources

Esempi correlati:

- AWS Organizations Tag Policy Syntax and Examples

Strumenti correlati:

- AWS Tag Editor

SEC07-BP02 Applicazione di controlli di protezione dei dati in base alla loro sensibilità

Applica controlli di protezione dei dati che forniscano un livello di controllo appropriato per ogni classe di dati definita nella tua policy di classificazione. Questa pratica può consentire di proteggere i dati sensibili dall'accesso e dall'uso non autorizzati, preservandone al contempo la disponibilità e l'utilizzo.

Risultato desiderato: hai una policy di classificazione che definisce i diversi livelli di sensibilità per i dati nella tua organizzazione. Per ciascuno di questi livelli di sensibilità, sono state pubblicate linee guida chiare per i servizi e i luoghi di archiviazione e movimentazione approvati e per la loro configurazione richiesta. I controlli per ogni livello vengono implementati in base al livello di protezione richiesto e ai costi associati. Disponi di un sistema di monitoraggio e di allerta per rilevare la presenza di dati in luoghi non autorizzati, l'elaborazione in ambienti non autorizzati, l'accesso da parte di soggetti non autorizzati o la configurazione di servizi correlati non conformi.

Anti-pattern comuni:

- Applicazione dello stesso livello di controlli di protezione su tutti i dati. Ciò può portare a un eccesso di controlli di sicurezza per i dati a bassa sensibilità o a una protezione insufficiente dei dati altamente sensibili.
- Non coinvolgere le parti interessate dei team di sicurezza, conformità e business nella definizione dei controlli sulla protezione dei dati.
- Trascurare le spese generali e i costi operativi associati all'implementazione e al mantenimento dei controlli sulla protezione dei dati.
- Non condurre revisioni periodiche del controllo della protezione dei dati per mantenere l'allineamento con le policy di classificazione.

Vantaggi dell'adozione di questa best practice: allineando i controlli al livello di classificazione dei dati, l'organizzazione può investire in livelli di controllo più elevati laddove necessario. Questo può includere l'aumento delle risorse per la sicurezza, il monitoraggio, la misurazione, la correzione e la rendicontazione. Se i controlli sono meno numerosi, è possibile migliorare l'accessibilità e la completezza dei dati per il personale, i clienti o gli utenti. Questo approccio offre alla tua organizzazione la massima flessibilità nell'utilizzo dei dati, pur rispettandone i requisiti di protezione.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

L'implementazione dei controlli di protezione dei dati in base ai loro livelli di sensibilità comporta diverse fasi fondamentali. Innanzitutto, consente di identificare i diversi livelli di sensibilità dei dati all'interno dell'architettura del tuo carico di lavoro (ad esempio, pubblico, interno, riservato e limitato) e di valutare il luogo in cui memorizzi ed elabori questi dati. Successivamente, definisci i limiti di isolamento dei dati in base al loro livello di sensibilità. Ti consigliamo di separare i dati in diversi Account AWS, utilizzando le policy di controllo dei servizi per limitare i servizi e le azioni consentite per ogni livello di sensibilità dei dati. In questo modo, puoi creare forti limiti di isolamento e far rispettare il principio del privilegio minimo.

Dopo aver definito i limiti di isolamento, implementa i controlli di protezione appropriati in base ai loro livelli di sensibilità. Fai riferimento alle best practice per la protezione dei dati a riposo e la protezione dei dati in transito per implementare controlli pertinenti come crittografia, controlli di accesso e audit. Prendi in considerazione tecniche come la tokenizzazione o l'anonimizzazione per ridurre il livello di sensibilità dei tuoi dati. Semplifica l'applicazione di policy coerenti sui dati in tutta l'azienda con un sistema centralizzato per la tokenizzazione e la de-tokenizzazione.

Monitora e verifica continuamente l'efficacia dei controlli implementati. Rivedi e aggiorna regolarmente lo schema di classificazione dei dati, le valutazioni dei rischi e i controlli di protezione in base all'evoluzione del panorama dei dati e delle minacce dell'organizzazione. Allinea i controlli di protezione dei dati implementati con le normative, gli standard e i requisiti legali pertinenti del settore. Inoltre, fornisci consapevolezza e formazione sulla sicurezza per aiutare i dipendenti a comprendere lo schema di classificazione dei dati e le loro responsabilità nella gestione e protezione dei dati sensibili.

Passaggi dell'implementazione

1. Identifica i livelli di classificazione e sensibilità dei dati all'interno del tuo carico di lavoro.
2. Definisci i limiti di isolamento per ogni livello e determina una strategia di applicazione.
3. Valuta i controlli definiti che regolano l'accesso, la crittografia, la verifica, la conservazione e altri aspetti richiesti dalla policy di classificazione dei dati.
4. Valuta le opzioni per ridurre il livello di sensibilità dei dati laddove appropriato, ad esempio utilizzando la tokenizzazione o l'anonimizzazione.
5. Verifica i tuoi controlli utilizzando test e monitoraggio automatici delle risorse configurate.

Risorse

Best practice correlate:

- PERF03-BP01 Uso di un archivio dati dedicato che supporta al meglio i requisiti di accesso e archiviazione dei dati
- COST04-BP05 Applicare policy di conservazione dei dati

Documenti correlati:

- Whitepaper sulla classificazione dei dati
- Best practice per la sicurezza, l'identità e la conformità
- AWS KMS Best Practices
- Encryption best practices and features for AWS services

Esempi correlati:

- Building a serverless tokenization solution to mask sensitive data

- How to use tokenization to improve data security and reduce audit scope

Strumenti correlati:

- AWS Key Management Service (AWS KMS)
- AWS CloudHSM
- AWS Organizations

SEC07-BP03 Automazione dell'identificazione e della classificazione

automatizzare l'identificazione e la classificazione dei dati può aiutarti a implementare i controlli corretti. L'uso dell'automazione per aumentare la determinazione manuale riduce il rischio di errore umano e di esposizione.

Risultato desiderato: è possibile verificare l'esistenza di controlli adeguati in base alla propria policy di classificazione e gestione. Gli strumenti e i servizi automatizzati ti aiutano a identificare e classificare il livello di sensibilità dei tuoi dati. L'automazione consente inoltre di monitorare continuamente gli ambienti per rilevare e avvisare se i dati vengono archiviati o gestiti in modo non autorizzato, in modo da poter intraprendere rapidamente azioni correttive.

Anti-pattern comuni:

- Affidarsi esclusivamente a processi manuali per l'identificazione e la classificazione dei dati, che possono essere soggetti a errori e richiedere tempi di lavoro lunghi. Questo può portare a una classificazione dei dati inefficiente e incoerente, soprattutto con l'aumento dei volumi di dati.
- Non disporre di meccanismi per tracciare e gestire le risorse di dati all'interno dell'organizzazione.
- Trascurare la necessità di un monitoraggio e di una classificazione continui dei dati durante i loro spostamenti e le loro trasformazioni all'interno dell'organizzazione.

Vantaggi dell'adozione di questa best practice: l'automazione dell'identificazione e della classificazione dei dati può portare a un'applicazione più coerente e accurata dei controlli sulla loro protezione, riducendo il rischio di errori umani. L'automazione può anche fornire visibilità sull'accesso e sul movimento dei dati sensibili, consentendo di rilevare le manipolazioni non autorizzate e a intraprendere azioni correttive.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Sebbene il giudizio umano sia spesso utilizzato per classificare i dati durante le fasi iniziali di progettazione di un carico di lavoro, è opportuno considerare la presenza di sistemi che automatizzino l'identificazione e la classificazione dei dati di test come controllo preventivo. Ad esempio, agli sviluppatori può essere fornito uno strumento o un servizio per analizzare i dati rappresentativi e determinarne la sensibilità. In AWS, è possibile caricare i set di dati in Amazon S3 e scansionarli utilizzando Amazon Macie, Amazon Comprehend o Amazon Comprehend Medical. Allo stesso modo, considera la scansione dei dati come parte dei test di unità e integrazione per individuare i casi in cui i dati sensibili non sono previsti. La segnalazione di dati sensibili in questa fase può evidenziare le lacune nelle protezioni prima dell'implementazione in produzione. Altre funzionalità, come il rilevamento di dati sensibili in AWS Glue, Amazon SNS e Amazon CloudWatch possono essere utilizzate anche per rilevare informazioni personali e intraprendere azioni di mitigazione. Per qualsiasi strumento o servizio automatizzato, capire come definisce i dati sensibili e integrare con altre soluzioni umane o automatizzate per colmare eventuali lacune.

Come controllo investigativo, utilizza il monitoraggio continuo degli ambienti per rilevare se i dati sensibili vengono archiviati in modi non conformi. Questo può aiutare a rilevare situazioni come l'emissione di dati sensibili nei file di log o la loro copia in un ambiente di analisi dei dati senza un'adeguata de-identificazione o redazione. I dati archiviati in Amazon S3 possono essere costantemente monitorati per verificare la presenza di dati sensibili grazie ad Amazon Macie.

Passaggi dell'implementazione

1. Eseguire una scansione iniziale degli ambienti per l'identificazione e la classificazione automatica.
    a. Una prima scansione completa dei dati può aiutare a capire dove risiedono i dati sensibili nei tuoi ambienti. Qualora una scansione completa non sia inizialmente richiesta o non possa essere completata in anticipo a causa dei costi, valuta se le tecniche di campionamento dei dati sono adatte a raggiungere i tuoi risultati. Ad esempio, Amazon Macie può essere configurato per eseguire un'ampia operazione automatizzata di rilevamento dei dati sensibili nei bucket S3. Questa funzionalità utilizza tecniche di campionamento per eseguire in modo efficiente in termini di costi un'analisi preliminare della posizione dei dati sensibili. È quindi possibile eseguire un'analisi più approfondita dei bucket S3 utilizzando un job di rilevamento dei dati sensibili. Anche altri archivi di dati possono essere esportati su S3 per essere analizzati da Macie.
2. Configurare scansioni continue dei tuoi ambienti.

   a. La capacità di rilevamento automatico dei dati sensibili di Macie può essere utilizzata per eseguire scansioni continue degli ambienti. I bucket S3 noti che sono autorizzati a memorizzare dati sensibili possono essere esclusi utilizzando un elenco di permessi in Macie.
3. Incorpora l'identificazione e la classificazione nei processi di compilazione e di test.
   a. Identifica gli strumenti che gli sviluppatori possono utilizzare per analizzare i dati alla ricerca di sensibilità mentre i carichi di lavoro sono in fase di sviluppo. Utilizza questi strumenti come parte dei test di integrazione per avvisare quando i dati sensibili sono inaspettati e impedire un'ulteriore distribuzione.
4. Implementa un sistema o un runbook per intervenire quando i dati sensibili vengono trovati in luoghi non autorizzati.

Risorse

Documenti correlati:

- AWS Glue: Detect and process sensitive data
- Using managed data identifiers in Amazon SNS
- Amazon CloudWatch Logs: Help protect sensitive log data with masking

Esempi correlati:

- Enabling data classification for Amazon RDS database with Macie
- Detecting sensitive data in DynamoDB with Macie

Strumenti correlati:

- Amazon Macie
- Amazon Comprehend
- Amazon Comprehend Medical
- AWS Glue

SEC07-BP04 Definizione della gestione del ciclo di vita dei dati scalabili

Comprendi i requisiti del ciclo di vita dei dati in relazione ai loro diversi livelli di classificazione e gestione. Si tratta di capire come vengono gestiti i dati quando entrano per la prima volta nel

tuo ambiente, come vengono trasformati e quali sono le regole per la loro distruzione. Prendi in considerazione fattori come i periodi di conservazione, l'accesso, il controllo e il monitoraggio della provenienza.

Risultato desiderato: classificare i dati il più vicino possibile al momento e al punto in cui vengono importati nel sistema. Quando la classificazione dei dati richiede il mascheramento, la tokenizzazione o altri processi che riducono il livello di sensibilità, si eseguono queste azioni il più vicino possibile al punto e al momento dell'importazione.

I dati vengono cancellati in conformità con la policy in uso quando non è più opportuno conservarli, in base alla loro classificazione.

Anti-pattern comuni:

- Implementare un approccio unico alla gestione del ciclo di vita dei dati, senza considerare i diversi livelli di sensibilità e i requisiti di accesso.
- Considerare la gestione del ciclo di vita solo dal punto di vista dei dati utilizzabili o dei dati di cui si esegue il backup, ma non di entrambi.
- Presumere che i dati immessi nel carico di lavoro siano validi, senza stabilirne il valore o la provenienza.
- Affidarsi alla durabilità dei dati come sostituti dei backup e della protezione dei dati.
- Conservare i dati oltre la loro utilità e il periodo di conservazione richiesto.

Vantaggi derivanti dall'adozione di questa best practice: una strategia di gestione del ciclo di vita dei dati ben definita e scalabile aiuta a mantenere la conformità alle normative, a migliorare la sicurezza dei dati, a ottimizzare i costi di archiviazione e a consentire un accesso e una condivisione efficienti, mantenendo al contempo controlli adeguati.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I dati all'interno di un carico di lavoro sono spesso dinamici. La forma che assumono quando entrano nell'ambiente del carico di lavoro può essere diversa da quella che assumono quando vengono archiviati o utilizzati nella logica aziendale, nel reporting, nell'analisi o nell'apprendimento automatico. Inoltre, il valore dei dati può cambiare nel tempo. Alcuni dati sono di natura temporale e perdono valore con il passare del tempo. Considera l'impatto di queste modifiche ai dati sulla valutazione del tuo schema di classificazione dei dati e dei controlli associati. Laddove possibile, utilizza un

meccanismo di ciclo di vita automatizzato, ad esempio le policy del ciclo di vita Amazon S3 e il Amazon Data Lifecycle Manager, per configurare i processi di conservazione, archiviazione e scadenza dei dati.

Distingui tra i dati disponibili per l'uso e quelli archiviati come backup. Prendi in considerazione l'utilizzo di AWS Backup per automatizzare il backup dei dati tra i servizi AWS. Le istantanee Amazon EBS consentono di copiare un volume EBS e archiviarlo utilizzando le funzionalità S3, inclusi il ciclo di vita, la protezione dei dati e l'accesso ai meccanismi di protezione. Due di questi meccanismi sono S3 Object Lock e AWS Backup Vault Lock, che possono fornire maggiore sicurezza e controllo sui backup. Gestisci una chiara separazione dei compiti e dell'accesso per i backup. Isola i backup a livello di account per mantenere la separazione dall'ambiente interessato durante un evento.

Un altro aspetto della gestione del ciclo di vita è la registrazione della cronologia dei dati man mano che avanzano nel carico di lavoro, chiamata tracciamento della provenienza dei dati. In questo modo hai la certezza di conoscere la provenienza dei dati, le trasformazioni effettuate, il proprietario o il processo che ha apportato le modifiche e la data. Questa cronologia è utile per la risoluzione dei problemi e le analisi in caso di potenziali eventi di sicurezza. Ad esempio, puoi registrare i metadati sulle trasformazioni in una tabella Amazon DynamoDB. All'interno di un data lake, puoi conservare copie dei dati trasformati in diversi bucket S3 per ogni fase della pipeline di dati. Memorizza le informazioni sullo schema e sul timestamp in un file AWS Glue Data Catalog. Indipendentemente dalla tua soluzione, considera i requisiti degli utenti finali per determinare gli strumenti appropriati di cui hai bisogno per segnalare la provenienza dei tuoi dati. Questo ti aiuterà a determinare come tracciare al meglio la tua provenienza.

Passaggi dell'implementazione

1. Analizza i tipi di dati, i livelli di sensibilità e i requisiti di accesso del carico di lavoro per classificare i dati e definire strategie di gestione del ciclo di vita appropriate.
2. Progetta e implementa policy di conservazione dei dati e processi di distruzione automatizzata in linea con i requisiti legali, normativi e organizzativi.
3. Stabilisci processi e automazione per il monitoraggio continuo, la verifica e l'adeguamento delle strategie, dei controlli e delle politiche di gestione del ciclo di vita dei dati in base all'evoluzione dei requisiti del carico di lavoro e delle normative.

Risorse

Best practice correlate:

- COST04-BP05 Applicare policy di conservazione dei dati
- SUS04-BP03 Utilizzo delle policy per gestire il ciclo di vita dei set di dati

Documenti correlati:

- Whitepaper sulla classificazione dei dati
- AWS Blueprint for Ransomware Defense
- DevOps Guidance: Improve traceability with data provenance tracking

Esempi correlati:

- How to protect sensitive data for its entire lifecycle in AWS
- Build data lineage for data lakes using AWS Glue, Amazon Neptune, and Spline

Strumenti correlati:

- AWS Backup
- Amazon Data Lifecycle Manager
- AWS Identity and Access Management Access Analyzer

SEC 8. In che modo proteggi i dati inattivi?

Proteggi i dati inattivi implementando più controlli, per ridurre il rischio di accessi non autorizzati o altri comportamenti impropri.

Best practice

- SEC08-BP01 Implementazione della gestione sicura delle chiavi
- SEC08-BP02 Applicazione della crittografia dei dati inattivi
- SEC08-BP03 Automatizzazione della protezione dei dati a riposo
- SEC08-BP04 Applicazione del controllo degli accessi

SEC08-BP01 Implementazione della gestione sicura delle chiavi

La gestione sicura delle chiavi include l'archiviazione, la rotazione, il controllo degli accessi e il monitoraggio del materiale relativo alla chiave necessario per proteggere i dati a riposo per il carico di lavoro.

Risultato desiderato: Un meccanismo di gestione delle chiavi dimensionabile, ripetibile e automatizzato. Il meccanismo dovrebbe fornire la possibilità di applicare l'accesso con il privilegio minimo al materiale relativo alla chiave e offrire il giusto equilibrio tra disponibilità, riservatezza e integrità delle chiavi. L'accesso alle chiavi deve essere monitorato e il materiale relativo alla chiave deve essere ruotato utilizzando un processo automatizzato. Il materiale relativo alla chiave non dovrebbe mai essere accessibile alle identità umane.

Anti-pattern comuni:

- Accesso umano a materiale relativo alla chiave non crittografato.
- Creazione di algoritmi crittografici personalizzati.
- Autorizzazioni di accesso al materiale relativo alla chiave troppo ampie.

Vantaggi dell'adozione di questa best practice: Attraverso un meccanismo di gestione delle chiavi sicuro per il tuo carico di lavoro, puoi contribuire a proteggere i contenuti dagli accessi non autorizzati. Inoltre, la crittografia dei dati potrebbe essere prevista da requisiti normativi per la tua organizzazione. Un'efficace soluzione di gestione delle chiavi può fornire meccanismi tecnici finalizzati alla protezione del materiale relativo alle chiavi in linea con tali normative.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Molti requisiti normativi e best practice includono la crittografia dei dati a riposo come controllo di sicurezza fondamentale. Per garantire la conformità, il carico di lavoro necessita di un meccanismo per archiviare e gestire in modo sicuro il materiale relativo alla chiave utilizzato per crittografare i dati a riposo.

AWS offre AWS Key Management Service (AWS KMS) per fornire uno spazio di archiviazione durevole, sicuro e ridondante per le chiavi AWS KMS. Molti servizi AWS si integrano con AWS KMS per supportare la crittografia dei dati. AWS KMS utilizza moduli di sicurezza hardware conformi allo standard FIPS 140-2 di livello 3 per proteggere le chiavi. Non esiste un meccanismo per esportare le chiavi AWS KMS convertendole in testo semplice.

Quando si distribuiscono carichi di lavoro utilizzando una strategia multi-account, una best practice è quella di mantenere le chiavi AWS KMS nello stesso account del carico di lavoro che le utilizza. In questo modello distribuito, la responsabilità della gestione delle chiavi AWS KMS spetta al team applicativo. In altri casi d'uso, le organizzazioni possono scegliere di archiviare le chiavi AWS KMS in un account centralizzato. Questa struttura centralizzata richiede policy aggiuntive per consentire l'accesso multi-account richiesto affinché l'account del carico di lavoro possa accedere alle chiavi archiviate nell'account centralizzato, ma può essere più applicabile nei casi d'uso in cui una singola chiave è condivisa tra Account AWS multipli.

Indipendentemente dalla posizione in cui è archiviato il materiale relativo alla chiave, l'accesso alla chiave deve essere strettamente controllato mediante l'uso di policy delle chiavi e policy IAM. Le policy delle chiavi costituiscono la modalità principale per controllare l'accesso a una chiave AWS KMS. Inoltre, AWS KMS garantisce che le chiavi possano fornire l'accesso ai servizi AWS per crittografare e decrittografare i dati per conto dell'utente. Prenditi del tempo per rivedere le best practice per il controllo degli accessi alle chiavi AWS KMS.

Una best practice è quella di monitorare l'uso delle chiavi di crittografia per rilevare modelli di accesso insoliti. Le operazioni eseguite utilizzando chiavi gestite da AWS e chiavi gestite dal cliente archiviate in AWS KMS, possono essere registrate in AWS CloudTrail e devono essere riviste periodicamente. Occorre prestare particolare attenzione al monitoraggio dei principali eventi di eliminazione delle chiavi. Per ridurre le probabilità di distruzione accidentale o dolosa del materiale relativo alla chiave, gli eventi di eliminazione delle chiavi non hanno efficacia immediata. I tentativi di eliminare le chiavi in AWS KMS sono soggetti a un periodo di attesa, che per impostazione predefinita è di 30 giorni, dando agli amministratori il tempo di rivedere queste azioni e annullare la richiesta, se necessario.

La maggior parte dei servizi AWS utilizza AWS KMS secondo una modalità chiara per te: il tuo unico requisito è decidere se utilizzare una chiave gestita da AWS o dal cliente. Se il carico di lavoro richiede l'uso diretto di AWS KMS per crittografare o decrittografare i dati, la best practice è utilizzare la crittografia a busta per proteggere i dati. Il comando SDK di crittografia AWS è in grado di fornire alle applicazioni primitive crittografiche lato client per implementare la crittografia a busta e integrarle con AWS KMS.

Passaggi dell'implementazione

1. Determina le opzioni di gestione della chiave appropriate (gestita da AWS o gestita dal cliente).
    - Per facilitare l'uso, AWS offre chiavi AWS di proprietà e gestite da AWS per la maggior parte dei servizi, fornendo funzionalità di crittografia a riposo senza la necessità di gestire il materiale o le policy delle chiavi.

- Quando utilizzi chiavi gestite dal cliente, prendi in considerazione il keystore predefinito per fornire il miglior equilibrio tra agilità, sicurezza, sovranità dei dati e disponibilità. Per altri casi d'uso può essere richiesto l'uso di archivi di chiavi personalizzati con AWS CloudHSM o di un archivio chiavi esterno.

2. Consulta l'elenco dei servizi che stai utilizzando per il tuo carico di lavoro per capire come AWS KMS si integra con il servizio. Ad esempio, le istanze EC2 possono utilizzare volumi EBS crittografati; verifica che anche le snapshot Amazon EBS create da tali volumi siano crittografate utilizzando una chiave gestita dal cliente e mitigando la divulgazione accidentale di dati di snapshot non crittografati.
   - Come i servizi AWS utilizzano AWS KMS
   - Per informazioni dettagliate sulle opzioni di crittografia offerte da un servizio AWS, consulta l'argomento Crittografia a riposo nella guida per l'utente o nella guida per sviluppatori del servizio.
3. Implementa AWS KMS: AWS KMS semplifica la creazione e la gestione delle chiavi e controlla l'uso della crittografia in un'ampia gamma di servizi AWS e nelle tue applicazioni.
   - Nozioni di base: AWS Key Management Service (AWS KMS)
   - Consulta le best practice per il controllo degli accessi alle chiavi AWS KMS.
4. Considera AWS Encryption SDK: utilizza l'AWS Encryption SDK con l'integrazione di AWS KMS quando la tua applicazione deve crittografare i dati lato client.
   - AWS Encryption SDK
5. Abilita IAM Access Analyzer per rivedere e inviare notifiche automaticamente se esistono policy delle chiavi AWS KMS eccessivamente permissive.
6. Abilita Security Hub per ricevere notifiche in caso di policy delle chiavi configurate in modo errato, chiavi programmate per essere eliminate o chiavi senza la rotazione automatica abilitata.
7. Determina il livello di log appropriato per le tue chiavi AWS KMS. Poiché le chiamate a AWS KMS, inclusi gli eventi di sola lettura, vengono registrate, i log CloudTrail associati a AWS KMS possono diventare voluminosi.
   - Alcune organizzazioni preferiscono separare l'attività di log di AWS KMS in un percorso separato. Per ulteriori informazioni, consulta la sezione Log delle chiamate API AWS KMS con CloudTrail della guida per gli sviluppatori AWS KMS.

Risorse

Documenti correlati:

- AWS Key Management Service
- Servizi e strumenti di crittografia di AWS
- Protezione dei dati Amazon S3 tramite la crittografia
- Envelope encryption
- Digital sovereignty pledge
- Demystifying AWS KMS key operations, bring your own key, custom key store, and ciphertext portability
- Dettagli di crittografia di AWS Key Management Service

Video correlati:

- Come funziona la crittografia in AWS
- Securing Your Block Storage on AWS (Protezione dello storage a blocchi in AWS).
- AWS data protection: Using locks, keys, signatures, and certificates

Esempi correlati:

- Implement advanced access control mechanisms using AWS KMS

SEC08-BP02 Applicazione della crittografia dei dati inattivi

Per i dati a riposo è necessario applicare la crittografia. La crittografia mantiene la riservatezza dei dati sensibili in caso di accesso non autorizzato o di divulgazione accidentale.

Risultato desiderato: la crittografia dei dati privati a riposo deve essere predefinita. La crittografia aiuta a mantenere la riservatezza dei dati e fornisce un ulteriore livello di protezione contro la divulgazione o esfiltrazione intenzionale o involontaria dei dati. I dati crittografati non possono essere letti o consultati senza che siano stati prima decrittografati. Tutti i dati archiviati in modo non crittografato devono essere inventariati e controllati.

Anti-pattern comuni:

- Mancato utilizzo di configurazioni con crittografia predefinita.
- Accesso estremamente permissivo alle chiavi di decrittografia.
- Mancato monitoraggio dell'uso delle chiavi di crittografia e decrittografia.

- Memorizzazione di dati non crittografati.
- Utilizzo della stessa chiave di crittografia per tutti i dati, indipendentemente dall'uso, dal tipo e dalla classificazione dei dati stessi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Mappa le chiavi di crittografia alle classificazioni dei dati all'interno dei carichi di lavoro. Questo approccio aiuta a proteggere dall'accesso estremamente permissivo quando si utilizza un'unica chiave di crittografia o un numero molto ridotto di chiavi di crittografia per i dati (consulta SEC07-BP01 Comprendere lo schema di classificazione dei dati).

AWS Key Management Service (AWS KMS) si integra con molti servizi AWS per semplificare la crittografia dei dati a riposo. Ad esempio, in Amazon Simple Storage Service (Amazon S3), puoi impostare la crittografia predefinita su un bucket in modo che i nuovi oggetti vengano automaticamente crittografati. Quando utilizzi AWS KMS, devi considerare il livello di restrizione dei dati. Le chiavi AWS KMS predefinite e controllate dal servizio sono gestite e utilizzate da AWS per tuo conto. Per i dati sensibili che richiedono un accesso granulare alla chiave di crittografia sottostante, è opportuno considerare le chiavi gestite dal cliente (CMK). L'utente ha il pieno controllo sulle CMK, anche per quanto riguarda la rotazione e la gestione degli accessi attraverso l'uso di policy sulle chiavi.

Inoltre, Amazon Elastic Compute Cloud (Amazon EC2) e Amazon S3 applicano la crittografia impostandone un tipo predefinito. Puoi servirti della Regole di AWS Config per verificare automaticamente l'utilizzo della crittografia, ad esempio, per volumi Amazon Elastic Block Store (Amazon EBS), istanze Amazon Relational Database Service (Amazon RDS) e bucket Amazon S3.

AWS offre anche soluzioni per la crittografia lato client, consentendo di crittografare i dati prima di caricarli nel cloud. AWS Encryption SDK offre un metodo per crittografare i dati utilizzando la crittografia a busta. L'utente fornisce la chiave di wrapping e AWS Encryption SDK genera una chiave dati unica per ogni oggetto di dati che crittografa. Considera AWS CloudHSM se hai bisogno di un modulo di sicurezza hardware (HSM) gestito single-tenant. AWS CloudHSM consente di generare, importare e gestire le chiavi crittografiche su un HSM convalidato FIPS 140-2 di livello 3. Alcuni casi d'uso di AWS CloudHSM includono la protezione delle chiavi private per il rilascio di un'autorità di certificazione (CA) e l'abilitazione della crittografia trasparente dei dati (TDE) per i database Oracle. Il client SDK AWS CloudHSM fornisce un software che consente di crittografare i dati sul lato client utilizzando le chiavi memorizzate all'interno di AWS CloudHSM prima di caricare i dati in AWS. La

Amazon DynamoDB Encryption Client consente inoltre di crittografare e firmare gli elementi prima del caricamento in una tabella DynamoDB.

Passaggi dell'implementazione

- Applicazione della crittografia a riposo per Amazon S3: implementa la crittografia predefinita del bucket Amazon S3.

  Configura la crittografia predefinita per i nuovi volumi Amazon EBS: specifica se desideri che tutti i nuovi volumi Amazon EBS vengano creati in forma crittografata, con la possibilità di utilizzare la chiave predefinita fornita da AWS o una chiave creata dall'utente.

  Configura Amazon Machine Image (AMI) crittografate: copiando un'AMI esistente con crittografia abilitata verrà eseguita la crittografia automatica di volumi root e delle snapshot.

  Configura la crittografia Amazon RDS: configura la crittografia per i cluster di database Amazon RDS e le snapshot a riposo utilizzando l'opzione di crittografia.

  Crea e configura le chiavi AWS KMS con policy che limitino l'accesso ai principali appropriati per ogni classificazione di dati: ad esempio, crea una chiave AWS KMS per la crittografia dei dati di produzione e una chiave diversa per la crittografia dei dati di sviluppo o di test. Puoi anche fornire l'accesso alle chiavi ad altri Account AWS. Considera la possibilità di avere account diversi per gli ambienti di sviluppo e di produzione. Qualora il tuo ambiente di produzione richieda la decodifica degli artefatti nell'account di sviluppo, puoi modificare la policy CMK utilizzata per crittografare gli artefatti di sviluppo per dare all'account di produzione la possibilità di decrittografare tali artefatti. L'ambiente di produzione può quindi importare i dati decrittografati per utilizzarli nella produzione.

  Configura la crittografia in altri servizi AWS: per gli altri servizi AWS utilizzati, consulta la documentazione sulla sicurezza del servizio per individuare le opzioni di crittografia del servizio.

Risorse

Documenti correlati:

- AWS Crypto Tools
- Documentazione di AWS
- AWS Encryption SDK
- AWS KMS Cryptographic Details Whitepaper (Whitepaper sui dettagli crittografici di AWS KMS)
- AWS Key Management Service

- AWS cryptographic services and tools (servizi e strumenti di crittografia AWS)
- Crittografia Amazon EBS
- Default encryption for Amazon EBS volumes (Crittografia predefinita per i volumi Amazon EBS)
- Crittografia delle risorse Amazon RDS
- How do I enable default encryption for an Amazon S3 bucket? (Come si attiva la crittografia predefinita per un bucket Amazon S3?)
- Protecting Amazon S3 Data Using Encryption (Protezione dei dati Amazon S3 mediante crittografia)

Video correlati:

- How Encryption Works in AWS (Come funziona la crittografia in AWS)
- Securing Your Block Storage on AWS (Protezione dello storage a blocchi in AWS)

SEC08-BP03 Automatizzazione della protezione dei dati a riposo

Usa l'automazione per convalidare e applicare i controlli dei dati a riposo. Usa la scansione automatica per rilevare le configurazioni errate delle soluzioni di archiviazione dei dati ed esegui le correzioni attraverso la risposta programmatica automatica, ove possibile. Incorpora l'automazione nei tuoi processi CI/CD per rilevare le configurazioni errate dell'archiviazione di dati prima che vengano implementate in produzione.

Risultato desiderato: i sistemi automatici scansionano e monitorano le posizioni di archiviazione di dati per individuare configurazioni errate di controlli, accessi non autorizzati e utilizzi imprevisti. Il rilevamento di posizioni di archiviazione non configurate avvia correzioni automatiche. I processi automatizzati creano backup dei dati e archiviano copie immutabili al di fuori dell'ambiente originale.

Anti-pattern comuni:

- Non considerare le opzioni per abilitare la crittografia dalle impostazioni predefinite, ove supportate.
- Non considerare gli eventi di sicurezza, oltre a quelli operativi, quando si formula una strategia di backup e ripristino automatizzata.
- Non applicare le impostazioni di accesso pubblico per i servizi di archiviazione.
- Non monitorare e verificare i controlli per proteggere i dati a riposo.

Vantaggi derivanti dall'adozione di questa best practice: l'automazione aiuta a prevenire il rischio di una configurazione errata delle posizioni di archiviazione di dati. Aiuta a prevenire che le configurazioni errate entrino negli ambienti di produzione. Questa best practice aiuta anche a rilevare e correggere eventuali configurazioni errate.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

L'automazione è un tema ricorrente in tutte le pratiche per la protezione dei dati a riposo. SEC01-BP06 Implementazione automatizzata dei controlli di sicurezza standard descrive come acquisire la configurazione delle risorse utilizzando modelli IaC (Infrastructure as code), ad esempio con AWS CloudFormation. Questi modelli sono vincolati a un sistema di controllo della versione e vengono utilizzati per distribuire risorse su AWS tramite una pipeline CI/CD. Queste tecniche si applicano anche all'automazione della configurazione delle soluzioni di archiviazione di dati, come le impostazioni di crittografia sui bucket Amazon S3.

Puoi controllare le impostazioni che definisci nei tuoi modelli IaC per eventuali configurazioni errate nelle pipeline CI/CD utilizzando le regole in AWS CloudFormation Guard. È possibile monitorare le impostazioni che non sono ancora disponibili in CloudFormation o in altri strumenti IaC per eventuali configurazioni errate con AWS Config. Gli avvisi generati da Config per configurazioni errate possono essere corretti automaticamente, come descritto in SEC04-BP04 Avvio della riparazione per le risorse non conformi.

L'utilizzo dell'automazione come parte della strategia di gestione delle autorizzazioni è anche parte integrante delle protezioni automatizzate dei dati. SEC03-BP02 Concessione dell'accesso con privilegio minimo e SEC03-BP04 Riduzione delle autorizzazioni in modo continuo descrivono continuamente la configurazione delle policy di accesso con privilegio minimo che vengono continuamente monitorate dal Sistema di AWS Identity and Access Management Access Analyzer per generare risultati quando è possibile ridurre l'autorizzazione. Oltre all'automazione per il monitoraggio delle autorizzazioni, puoi configurare Amazon GuardDuty per monitorare comportamenti anomali di accesso ai dati per i tuoi volumi EBS (tramite un'istanza EC2), i bucket S3 e i database Amazon Relational Database Service supportati.

L'automazione svolge anche la funzione di rilevare quando i dati sensibili vengono archiviati in luoghi non autorizzati. SEC07-BP03 Automazione dell'identificazione e della classificazione descrive come Amazon Macie può monitorare i bucket S3 per la ricerca di dati sensibili imprevisti e generare avvisi in grado di avviare una risposta automatica.

Segui le procedure descritte in REL09 Backup dei dati per sviluppare una strategia automatizzata di backup e ripristino dei dati. Il backup e il ripristino dei dati sono importanti tanto per il ripristino da eventi di sicurezza quanto per gli eventi operativi.

Passaggi dell'implementazione

1. Acquisisci la configurazione dell'archiviazione di dati nei modelli IaC. Utilizza i controlli automatici nelle pipeline CI/CD per rilevare configurazioni errate.
   a. Puoi utilizzare <ulink type="marketing" url="cloudformation">&CFN;</ulink> per i tuoi modelli IaC e CloudFormation Guard per verificare la presenza di errori di configurazione nei modelli.
   b. Utilizza AWS Config per eseguire le regole in modalità di valutazione proattiva. Utilizza questa impostazione per verificare la conformità di una risorsa come passaggio della pipeline CI/CD prima di crearla.
2. Monitora le risorse per individuare eventuali configurazioni errate dell'archiviazione di dati.
   a. Imposta AWS Config per monitorare le risorse di archiviazione di dati per eventuali modifiche nelle configurazioni di controllo e generare avvisi per richiamare azioni correttive quando rileva una configurazione errata.
   b. Vedi SEC04-BP04 Avvio della riparazione delle risorse non conformi per ulteriori indicazioni sulle correzioni automatiche.
3. Monitora e riduci continuamente le autorizzazioni di accesso ai dati tramite l'automazione.
   a. IAM Access Analyzer può essere eseguito continuamente per generare avvisi quando le autorizzazioni possono essere potenzialmente ridotte.
4. Monitora e avvisa in caso di comportamenti anomali di accesso ai dati.
   a. GuardDuty controlla sia le firme delle minacce note sia le deviazioni dai comportamenti di accesso di base per le risorse di archiviazione dei dati come i volumi EBS, i bucket S3 e i database RDS.
5. Monitora e invia avvisi sui dati sensibili archiviati in luoghi inaspettati.
   a. Utilizza Amazon Macie per scansionare continuamente i tuoi bucket S3 alla ricerca di dati sensibili.
6. Automatizza i backup sicuri e crittografati dei tuoi dati.
   a. AWS Backup è un servizio gestito che crea backup crittografati e sicuri di varie origini dati su AWS. Elastic Disaster Recovery consente di copiare carichi di lavoro completi del server e mantenere una protezione continua dei dati con un obiettivo del punto di ripristino (RPO) misurato in secondi. È possibile configurare entrambi i servizi in modo che lavorino insieme per

automatizzare la creazione di backup dei dati e la loro copia in posizioni di failover. Questo può aiutare a mantenere i dati disponibili in caso di eventi operativi o di sicurezza.

Risorse

Best practice correlate:

- SEC01-BP06 Implementazione automatizzata dei controlli di sicurezza standard
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- SEC03-BP04 Riduzione delle autorizzazioni in modo continuo
- SEC04-BP04 Avvio della riparazione delle risorse non conformi
- SEC07-BP03 Automazione dell'identificazione e della classificazione
- REL09-BP02 Protezione e crittografia dei backup
- REL09-BP03 Esecuzione del backup dei dati in automatico

Documenti correlati:

- AWS Prescriptive Guidance: Automatically encrypt existing and new Amazon EBS volumes
- Ransomware Risk Management on AWS Using the NIST Cyber Security Framework (CSF)

Esempi correlati:

- How to use AWS Config proactive rules and AWS CloudFormation Hooks to prevent creation of noncompliant cloud resources
- Automate and centrally manage data protection for Amazon S3 with AWS Backup
- AWS re:Invent 2023 - Implement proactive data protection using Amazon EBS snapshots
- AWS re:Invent 2022 - Build and automate for resilience with modern data protection

Strumenti correlati:

- AWS CloudFormation Guard
- AWS CloudFormation Guard Rules Registry
- IAM Access Analyzer
- Amazon Macie

- AWS Backup
- Ripristino di emergenza elastico

SEC08-BP04 Applicazione del controllo degli accessi

Per proteggere i dati a riposo, applica il controllo degli accessi utilizzando meccanismi come l'isolamento e il controllo delle versioni, quindi applica il principio del privilegio minimo. Impedisci l'accesso pubblico ai dati.

Risultato desiderato: verifica che solo gli utenti autorizzati possano accedere ai dati in base al principio "Need-to-Know" (necessità di sapere). La protezione dei dati è assicurata da backup regolari e dal controllo delle versioni, per evitare che i dati vengano modificati o eliminati intenzionalmente o inavvertitamente. L'isolamento dei dati critici dagli altri dati ne protegge la riservatezza e l'integrità.

Anti-pattern comuni:

- Archiviazione dei dati con requisiti di sensibilità o classificazione diversi.
- Utilizzo di autorizzazioni troppo permissive sulle chiavi di decrittografia.
- Classificazione impropria dei dati.
- Nessun mantenimento di backup dettagliati dei dati importanti.
- Accesso persistente ai dati di produzione.
- Nessun audit dell'accesso ai dati o revisione periodica delle autorizzazioni.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

La protezione dei dati a riposo può essere garantita da diversi controlli, tra cui l'accesso (utilizzando il privilegio minimo), l'isolamento e il controllo delle versioni. L'accesso ai dati deve essere soggetto a audit mediante meccanismi di rilevazione, come AWS CloudTrail e log sul livello di servizio, come i log di accesso di Amazon Simple Storage Service (Amazon S3). Per ridurre nel tempo la quantità di dati disponibili pubblicamente, è necessario fare un inventario dei dati accessibili pubblicamente e creare un piano.

Amazon S3 Glacier Vault Lock e Amazon S3 Object Lock forniscono un controllo di accesso obbligatorio per gli oggetti in Amazon S3: una volta bloccata con l'opzione di conformità, una policy Vault non può essere modificata nemmeno dall'utente root fino alla scadenza del blocco.

Passaggi dell'implementazione

- Applica il controllo degli accessi: applica il controllo degli accessi con privilegio minimo, incluso l'accesso alle chiavi di crittografia.
- Separa i dati in base a diversi livelli di classificazione: utilizza diversi Account AWS per i livelli di classificazione dei dati e gestisci tali account utilizzando AWS Organizations.
- Rivedi le policy di AWS Key Management Service (AWS KMS): rivedi il livello di accesso concesso nelle policy di AWS KMS.
- Rivedi le autorizzazioni dei bucket e degli oggetti di Amazon S3: rivedi regolarmente il livello di accesso concesso nelle policy dei bucket S3. La best practice è evitare di utilizzare bucket leggibili o scrivibili pubblicamente. Valuta l'utilizzo di AWS Config per rilevare i bucket disponibili pubblicamente e di Amazon CloudFront per fornire contenuti provenienti da Amazon S3. Verifica che i bucket che non consentono l'accesso pubblico siano configurati correttamente per impedirlo. Per impostazione predefinita, tutti i bucket S3 sono privati e possono accedervi soltanto gli utenti a cui è stato esplicitamente accordato l'accesso.
- Abilita AWS IAM Access Analyzer: IAM Access Analyzer analizza i bucket Amazon S3 e genera un risultato quando una policy S3 concede l'accesso a un'entità esterna.
- Abilita il controllo delle versioni Amazon S3 e del blocco degli oggetti laddove appropriato.
- UtilizzaAmazon S3 Inventory: Amazon S3 Inventory può essere utilizzato per effettuare audit e report sullo stato di replica e crittografia degli oggetti S3.
- Rivedi le autorizzazioni di condivisione Amazon EBS e AMI: le autorizzazioni di condivisione possono consentire la condivisione di immagini e volumi con Account AWS esterni al carico di lavoro.
- Rivedi periodicamente le condivisioni di AWS Resource Access Manager per stabilire se le risorse devono continuare ad essere condivise. Resource Access Manager consente di condividere risorse, come le policy del firewall di rete AWS, le regole del resolver Amazon Route 53 e le sottoreti, all'interno dei Amazon VPC. Sottoponi regolarmente a audit le risorse condivise e interrompi la condivisione delle risorse che non devono più essere condivise.

Risorse

Best practice correlate:

- SEC03-BP01 Definizione dei requisiti di accesso
- SEC03-BP02 Concessione dell'accesso con privilegio minimo

Documenti correlati:

- AWS KMS Cryptographic Details Whitepaper (Whitepaper sui dettagli crittografici di AWS KMS)
- Introduction to Managing Access Permissions to Your Amazon S3 Resources (Introduzione alla gestione delle autorizzazioni di accesso alle risorse di Amazon S3)
- Overview of managing access to your AWS KMS resources (Panoramica della gestione dell'accesso alle risorse AWS KMS)
- Regole di AWS Config (Regole AWS Config)
- Amazon S3 + Amazon CloudFront: A Match Made in the Cloud (Amazon S3 + Amazon CloudFront: un abbinamento perfetto nel cloud)
- Utilizzo del controllo delle versioni
- Utilizzo del blocco oggetti Amazon S3
- Condivisione di uno snapshot Amazon EBS
- AMI condivise
- Hosting a single-page application on Amazon S3 (Ospitare un'applicazione a pagina singola su Amazon S3)

Video correlati:

- Securing Your Block Storage on AWS (Protezione dello storage a blocchi in AWS)

SEC 9. In che modo proteggi i dati in transito?

Proteggi i dati in transito implementando più controlli, per ridurre il rischio di accessi non autorizzati o perdita.

Best practice

- SEC09-BP01 Implementazione della gestione sicura delle chiavi e dei certificati
- SEC09-BP02 Applicazione della crittografia dei dati in transito
- SEC09-BP03 Autenticazione delle comunicazioni di rete

SEC09-BP01 Implementazione della gestione sicura delle chiavi e dei certificati

I certificati Transport Layer Security (TLS) vengono utilizzati per proteggere le comunicazioni di rete e stabilire l'identità di siti web, risorse e carichi di lavoro su Internet, nonché sulle reti private.

Risultato desiderato: un sistema di gestione dei certificati sicuro in grado di fornire, implementare, archiviare e rinnovare i certificati in un'infrastruttura a chiave pubblica (PKI). Un meccanismo sicuro di gestione delle chiavi e dei certificati impedisce la divulgazione del materiale relativo alle chiavi private dei certificati e rinnova automaticamente il certificato su base periodica. Si integra inoltre con altri servizi per fornire comunicazioni di rete e identità sicure per le risorse delle macchine all'interno del carico di lavoro. Il materiale relativo alla chiave non dovrebbe mai essere accessibile alle identità umane.

Anti-pattern comuni:

- Esecuzione di passaggi manuali durante i processi di distribuzione, implementazione o rinnovo dei certificati.
- Attenzione insufficiente alla gerarchia delle autorità di certificazione (CA) durante la progettazione di una CA privata.
- Utilizzo di certificati autofirmati per risorse pubbliche.

Vantaggi dell'adozione di questa best practice:

- Semplificazione della gestione dei certificati attraverso la distribuzione, l'implementazione e il rinnovo automatizzati
- Incoraggiamento dell'utilizzo della crittografia dei dati in transito con l'utilizzo di certificati TLS
- Maggiore sicurezza e verificabilità delle operazioni di certificazione intraprese dall'autorità di certificazione
- Organizzazione delle mansioni di gestione ai diversi livelli della gerarchia della CA

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

I carichi di lavoro moderni fanno ampio uso di comunicazioni di rete crittografate utilizzando protocolli PKI come TLS. La gestione dei certificati PKI può essere complessa, ma la fornitura, la distribuzione, l'implementazione e il rinnovo automatizzati dei certificati possono ridurre l'attrito associato alla loro gestione.

AWS fornisce due servizi per gestire i certificati PKI generici: AWS Certificate Manager e AWS Private Certificate Authority (AWS Private CA). ACM è il servizio principale utilizzato dai clienti per fornire, gestire e implementare certificati da utilizzare sia in carichi di lavoro di AWS pubblici che

privati. ACM emette certificati utilizzando AWS Private CA e si integra con molti altri servizi AWS gestiti per fornire certificati TLS sicuri per i carichi di lavoro.

AWS Private CA consente di stabilire la propria autorità di certificazione principale o subordinata e di emettere certificati TLS tramite un'API. È possibile utilizzare questo tipo di certificati in scenari in cui si mantengono il controllo e la gestione della catena di fiducia sul lato client della connessione TLS. Oltre ai casi d'uso TLS, AWS Private CA può essere utilizzato per emettere certificati per i pod Kubernetes, gli attestati dei prodotti dei dispositivi Matter, la firma del codice e altri casi d'uso che prevedono un modello personalizzato. Puoi anche utilizzare la strategia di IAM Roles Anywhere per fornire credenziali temporanee IAM ai carichi di lavoro on-premise ai quali sono stati assegnati certificati X.509 firmati dalla tua CA privata.

Oltre a ACM e AWS Private CA, AWS IoT Core fornisce supporto specializzato per il provisioning, la gestione e l'implementazione di certificati PKI su dispositivi IoT. AWS IoT Core fornisce meccanismi specializzati per l'onboarding di dispositivi IoT nella tua infrastruttura a chiave pubblica su larga scala.

Considerazioni sulla creazione di una gerarchia CA privata

Quando è necessario stabilire una CA privata, è importante prestare particolare attenzione a progettare correttamente la gerarchia della CA fin dall'inizio. Quando si crea una gerarchia CA privata è consigliabile distribuire ogni livello della gerarchia CA su Account AWS separati. Questo passaggio intenzionale riduce l'estensione di ogni livello della gerarchia della CA, semplificando l'individuazione delle anomalie nei dati di log di CloudTrail e riducendo l'ambito di accesso o l'impatto in caso di accesso non autorizzato a uno degli account. La CA principale deve risiedere in un account separato e deve essere utilizzata solo per emettere uno o più certificati CA intermedi.

Quindi, crea una o più CA intermedie in account separati dall'account della CA principale per emettere certificati per utenti finali, dispositivi o altri carichi di lavoro. Infine, emetti certificati della tua CA principale a uso delle CA intermedie, che a loro volta emetteranno certificati per gli utenti finali o i dispositivi. Per ulteriori informazioni sulla pianificazione dell'implementazione della CA e sulla progettazione della gerarchia delle CA, inclusa la pianificazione della resilienza, la replica tra regioni, la condivisione delle CA all'interno dell'organizzazione e altro ancora, consulta Pianificazione dell'implementazione di AWS Private CA.

Passaggi dell'implementazione

1. Determina i servizi AWS pertinenti richiesti per il tuo caso d'uso:

- Molti casi d'uso possono sfruttare l'infrastruttura a chiave pubblica AWS esistente utilizzando AWS Certificate Manager. ACM può essere utilizzato per implementare certificati TLS per server Web, sistemi di bilanciamento del carico o altri usi per certificati pubblicamente affidabili.
- Considera il servizio AWS Private CA quando è necessario stabilire una gerarchia di autorità di certificazione privata o accedere a certificati esportabili. ACM può quindi essere utilizzato per emettere molti tipi di certificati dell'entità finale utilizzando AWS Private CA.
- Per i casi d'uso in cui i certificati devono essere forniti su larga scala per dispositivi Internet delle cose (IoT) integrati, prendi in considerazione l'uso di AWS IoT Core.

2. Implementa il rinnovo automatico dei certificati quando possibile:
   - utilizza rinnovo gestito di ACM per i certificati emessi da ACM insieme ai servizi AWS gestiti integrati.
3. Stabilisci percorsi di registrazione e controllo:
   - Abilita log CloudTrail per tenere traccia degli accessi agli account che detengono le autorità di certificazione. Prendi in considerazione la possibilità di configurare la convalida dell'integrità dei file di log in CloudTrail per verificarne l'autenticità dei dati.
   - Genera e rivedi periodicamente rapporti di audit che elencano i certificati che la tua CA privata ha emesso o revocato. Questi report possono essere esportati in un bucket S3.
   - Quando si implementa una CA privata, è inoltre necessario creare un bucket S3 per archiviare l'elenco di revoche dei certificati (CRL). Per indicazioni sulla configurazione di questo bucket S3 in base ai requisiti del carico di lavoro, consulta Pianificazione di un elenco di revoche di certificati (CRL).

Risorse

Best practice correlate:

- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC08-BP01 Implementazione della gestione sicura delle chiavi
- SEC09-BP03 Autenticazione delle comunicazioni di rete

Documenti correlati:

- Come ospitare e gestire un'intera infrastruttura di certificati privata in AWS
- How to secure an enterprise scale ACM Private CA hierarchy for automotive and manufacturing

- Private CA best practices
- How to use AWS RAM to share your ACM Private CA cross-account

Video correlati:

- Activating AWS Certificate Manager Private CA (workshop)

Esempi correlati:

- Private CA workshop
- Workshop sulla gestione dei dispositivi IOT (incluso il provisioning dei dispositivi)

Strumenti correlati:

- Plugin to Kubernetes cert-manager to use AWS Private CA

SEC09-BP02 Applicazione della crittografia dei dati in transito

Applica i requisiti di crittografia definiti in base alle policy, agli obblighi normativi e agli standard dell'organizzazione per contribuire a soddisfare i requisiti organizzativi, legali e di conformità. Utilizza solo protocolli con crittografia quando trasmetti dati sensibili al di fuori del tuo cloud privato virtuale (VPC). La crittografia aiuta a mantenere la riservatezza dei dati anche quando questi transitano su reti non affidabili.

Risultato desiderato: tutti i dati devono essere crittografati in transito utilizzando protocolli e suite di crittografia TLS sicuri. Il traffico di rete tra le tue risorse e Internet deve essere crittografato per evitare l'accesso non autorizzato ai dati. Il traffico di rete esclusivamente all'interno dell'ambiente AWS deve essere crittografato utilizzando TLS, ove possibile. La rete interna di AWS è crittografata per impostazione predefinita e il traffico di rete all'interno di un VPC non può essere sottoposto a spoofing o sniffing, a meno che una parte non autorizzata non abbia ottenuto l'accesso alla risorsa che sta generando il traffico (come le istanze Amazon EC2 e i container Amazon ECS). Considera la possibilità di proteggere il traffico da rete a rete con una rete privata virtuale (VPN) IPsec.

Anti-pattern comuni:

- Utilizzo di versioni obsolete di SSL, TLS e componenti della suite di crittografia (ad esempio, SSL v3.0, chiavi RSA a 1024 bit e crittografia RC4).

- Autorizzazione del traffico non criptato (HTTP) verso o da risorse pubbliche.
- Monitoraggio e sostituzione mancati dei certificati X.509 prima della scadenza.
- Utilizzo di certificati X.509 autofirmati per TLS.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I servizi AWS forniscono endpoint HTTPS utilizzando TLS per le comunicazioni e forniscono crittografia in transito quando comunicano con le API AWS. I protocolli non sicuri, come HTTP, possono essere sottoposti a audit e bloccati in un VPC tramite l'uso di gruppi di sicurezza. Le richieste HTTP possono essere reindirizzate automaticamente a HTTPS in Amazon CloudFront o su un Application Load Balancer. Hai il controllo completo sulle tue risorse informatiche per implementare la crittografia in transito nei tuoi servizi. Inoltre, puoi utilizzare la connettività VPN nel VPC da una rete esterna o AWS Direct Connect per facilitare la crittografia del traffico. Verifica che i tuoi client effettuino chiamate alle API AWS utilizzando almeno TLS 1.2, poiché AWS considererà obsoleto l'utilizzo di TLS 1.0 e 1.1 da giugno 2023. Per requisiti particolari, in Marketplace AWS sono disponibili soluzioni di terze parti.

Passaggi dell'implementazione

- Applicazione della crittografia in transito: i requisiti di crittografia definiti devono essere basati sugli standard e sulle best practice più recenti e consentire solo protocolli sicuri. Ad esempio, configura un gruppo di sicurezza per consentire solo il protocollo HTTPS a un Application Load Balancer o a un'istanza Amazon EC2.
- Configura i protocolli sicuri nei servizi edge: configura HTTPS con Amazon CloudFront e utilizza un profilo di sicurezza appropriato per la postura di sicurezza e il caso d'uso.
- Utilizza una VPN per la connettività esterna: valuta l'impiego di una VPN IPsec per la protezione delle connessioni punto a punto o rete a rete al fine di garantire la riservatezza e l'integrità dei dati.
- Configura protocolli sicuri nei sistemi di bilanciamento del carico: seleziona una policy di sicurezza che fornisca le suite di crittografia più efficaci supportate dai client che si connetteranno all'ascoltatore. Configurazione di un ascoltatore HTTPS per Application Load Balancer.
- Configura protocolli sicuri in Amazon Redshift: configura il cluster per richiedere una connessione Secure Socket Layer (SSL) o Transport Layer Security (TLS).
- Configura protocolli sicuri: analizza la documentazione relativa al servizio AWS per determinare le capacità di crittografia in transito.

- Configura l'accesso sicuro durante il caricamento di bucket Amazon S3: utilizza i controlli delle policy del bucket Amazon S3 per applicare l'accesso sicuro ai dati.
- Valuta l'utilizzo di AWS Certificate Manager: ACM consente di fornire, gestire e implementare certificati TLS pubblici da utilizzare con i servizi AWS.
- Valuta l'utilizzo di AWS Private Certificate Authority per esigenze di PKI private: AWS Private CA consente di creare gerarchie di autorità di certificazione (CA) private per emettere certificati X.509 end-entity che possono essere usati per creare canali TLS crittografati.

Risorse

Documenti correlati:

- Documentazione di AWS
- Utilizzo di HTTPS con CloudFront
- Connetti il tuo VPC a reti remote utilizzando AWS Virtual Private Network
- Configurazione di un ascoltatore HTTPS per Application Load Balancer
- Tutorial: configurazione di SSL/TLS su Amazon Linux 2 Amazon Linux 2
- Utilizzo di SSL/TLS per crittografare una connessione a un'istanza database
- Configurazione delle opzioni di sicurezza per le connessioni

SEC09-BP03 Autenticazione delle comunicazioni di rete

Verifica l'identità delle comunicazioni utilizzando protocolli che supportano l'autenticazione, ad esempio Transport Layer Security (TLS) o IPsec.

Progetta il carico di lavoro in modo da utilizzare protocolli di rete sicuri e autenticati per le comunicazioni tra servizi, applicazioni o utenti. L'utilizzo di protocolli di rete che supportano l'autenticazione e l'autorizzazione offre un controllo più rigido sui flussi di rete e riduce l'impatto di eventuali accessi non autorizzati.

Risultato desiderato: un carico di lavoro con flussi di traffico del piano dati e del piano di controllo (control-plane) ben definiti tra i servizi. I flussi di traffico utilizzano protocolli di rete autenticati e crittografati laddove tecnicamente fattibile.

Anti-pattern comuni:

- Flussi di traffico non crittografati o non autenticati all'interno del carico di lavoro.
- Riutilizzo delle credenziali di autenticazione tra più utenti o entità.
- Uso esclusivo di controlli di rete come meccanismo di controllo degli accessi.
- Creazione di un meccanismo di autenticazione personalizzato anziché usare meccanismi di autenticazione standard del settore.
- Flussi di traffico eccessivamente permissivi tra i componenti del servizio o altre risorse nel VPC.

Vantaggi dell'adozione di questa best practice:

- Limita l'ambito dell'impatto di eventuali accessi non autorizzati a una parte del carico di lavoro.
- Fornisce un livello più elevato di sicurezza affinché le azioni vengano eseguite solo da entità autenticate.
- Migliora il disaccoppiamento dei servizi definendo e applicando chiaramente le interfacce di trasferimento dei dati previste.
- Migliora il monitoraggio, la registrazione in log e la risposta agli incidenti tramite l'attribuzione delle richieste e interfacce di comunicazione ben definite.
- Fornisce un livello elevatissimo di difesa ai carichi di lavoro combinando i controlli di rete con i controlli di autenticazione e autorizzazione.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

I modelli di traffico di rete del carico di lavoro possono essere suddivisi in due categorie:

- Il traffico orizzontale (sinistra-destra) rappresenta i flussi di traffico tra servizi che costituiscono un carico di lavoro.
- Il traffico verticale (alto-basso) rappresenta i flussi di traffico tra il carico di lavoro e i consumatori.

Mentre crittografare il traffico verticale (alto-basso) è prassi comune, proteggere il traffico orizzontale (sinistra-destra) mediante protocolli autenticati non è così frequente. Le moderne best practice di sicurezza raccomandano che la progettazione della rete non sia l'unico elemento in grado di garantire una relazione affidabile tra due entità. Quando due servizi possono trovarsi all'interno di una rete comune, è comunque consigliabile crittografare, autenticare e autorizzare le comunicazioni tra tali servizi.

Ad esempio, le API del servizio AWS utilizzano il protocollo di firma AWS Signature Version 4 (SigV4) per autenticare il chiamante, indipendentemente dalla rete da cui proviene la richiesta. Questa autenticazione garantisce che le API AWS possano verificare l'identità che ha richiesto l'azione e che tale identità possa quindi essere combinata con le policy per decidere se autorizzare o meno l'azione.

Servizi come Amazon VPC Lattice e Amazon API Gateway consentono di utilizzare lo stesso protocollo di firma SigV4 per aggiungere funzionalità di autenticazione e autorizzazione al traffico orizzontale (sinistra-destra) ai carichi di lavoro. Se le risorse esterne all'ambiente AWS devono comunicare con servizi che richiedono l'autenticazione e l'autorizzazione basate su SigV4, è possibile utilizzare AWS Identity and Access Management (IAM) Roles Anywhere sulla risorsa AWS per acquisire credenziali AWS temporanee. Queste credenziali possono essere utilizzate per firmare richieste ai servizi che utilizzano SigV4 per autorizzare l'accesso.

Un altro meccanismo comune per l'autenticazione del traffico orizzontale (sinistra-destra) è l'autenticazione reciproca TLS (mTLS). Molte applicazioni Internet delle cose (IoT), business-to-business (B2B) e microservizi utilizzano mTLS per convalidare l'identità di entrambi i lati di una comunicazione TLS mediante l'uso di certificati X.509 lato client e lato server. Questi certificati possono essere emessi da AWS Private Certificate Authority (AWS Private CA). È possibile utilizzare servizi come Amazon API Gateway e AWS App Mesh per fornire l'autenticazione mTLS per la comunicazione tra carichi di lavoro a tutti i livelli. Sebbene fornisca informazioni di autenticazione per entrambi i lati di una comunicazione TLS, mTLS non fornisce un meccanismo di autorizzazione.

Infine, OAuth 2.0 e OpenID Connect (OIDC) sono due protocolli generalmente utilizzati per controllare l'accesso ai servizi da parte degli utenti, ma stanno diventando popolari anche per il traffico a livello di servizi. API Gateway fornisce un sistema di autorizzazione JSON Web Token (JWT), che consente ai carichi di lavoro di limitare l'accesso alle route API utilizzando JWT emessi da gestori dell'identità digitale OIDC o OAuth 2.0. Gli ambiti OAuth2 possono essere utilizzati come base per decisioni di autorizzazione essenziali, ma i controlli di autorizzazione devono comunque essere implementati a livello di applicazione. Gli ambiti OAuth2 da soli non possono supportare requisiti di autorizzazione più complessi.

Passaggi dell'implementazione

- Definisci e documenta i flussi di rete del carico di lavoro: il primo passo per implementare una strategia di difesa di alto profilo è definire i flussi di traffico del carico di lavoro.
  - Crea un diagramma del flusso di dati che definisca chiaramente come vengono trasmessi i dati tra i diversi servizi che costituiscono il carico di lavoro. Questo diagramma è il primo passo per autorizzare tali flussi nei canali di rete autenticati.

    - Nelle fasi di sviluppo e test dota il carico di lavoro di strumenti per controllare che il diagramma del flusso dei dati rifletta accuratamente il comportamento del carico di lavoro in fase di esecuzione.
    - Un diagramma del flusso dei dati può essere utile anche quando si esegue un esercizio di modellazione delle minacce, come descritto in SEC01-BP07 Identificare le minacce e dare priorità alle mitigazioni utilizzando un modello di minaccia.
- Definisci i controlli di rete: considera le funzionalità AWS per stabilire controlli di rete allineati ai flussi di dati. Sebbene i confini della rete non debbano costituire l'unico elemento di controllo della sicurezza, essi forniscono un livello nella strategia di difesa di alto profilo a protezione del carico di lavoro.
    - Utilizza i gruppi di sicurezza per stabilire, definire e limitare i flussi di dati tra risorse.
    - Valuta l'utilizzo di AWS PrivateLink per comunicare sia con AWS che con i servizi di terze parti che supportano AWS PrivateLink. I dati inviati tramite un endpoint di interfaccia AWS PrivateLink rimangono all'interno della dorsale della rete AWS e non attraversano la rete Internet pubblica.
- Implementa l'autenticazione e l'autorizzazione tra i servizi del carico di lavoro: scegli il set di servizi AWS più appropriato per fornire flussi di traffico autenticati e crittografati nel carico di lavoro.
    - Valuta l'ipotesi di utilizzare Amazon VPC Lattice per la sicurezza della comunicazione tra servizi. VPC Lattice può utilizzare l'autenticazione SigV4 combinata con le policy di autenticazione per controllare l'accesso a livello di servizi.
    - Per la comunicazione tra servizi tramite mTLS, valuta l'ipotesi di utilizzare API Gateway o App Mesh. AWS Private CA può essere utilizzato per stabilire una gerarchia di autorità di certificazione (CA) private in grado di emettere certificati da utilizzare con mTLS.
    - Quando esegui l'integrazione con servizi che utilizzano OAuth 2.0 o OIDC, considera l'utilizzo del sistema di autorizzazione JWT da parte di API Gateway.
    - Per la comunicazione tra il carico di lavoro e i dispositivi IoT, considera l'utilizzo di AWS IoT Core, che offre diverse opzioni per la crittografia e l'autenticazione del traffico di rete.
- Monitora gli accessi non autorizzati: monitora continuamente i canali di comunicazione non intenzionali, i responsabili non autorizzati che tentano di accedere alle risorse protette e altri schemi di accesso impropri.
    - In caso di utilizzo di VPC Lattice per gestire l'accesso ai servizi, valuta la possibilità di abilitare e monitorare i log di accesso di VPC Lattice. Questi log di accesso includono informazioni sull'entità richiedente, informazioni di rete tra cui VPC di origine e destinazione e metadati della richiesta.

- Valuta la possibilità di abilitare i log di flusso VPC per acquisire i metadati sui flussi di rete e verificare periodicamente la presenza di anomalie.
- Consulta il manuale AWSSecurity Incident Response Guide e la sezione relativa alle risposte agli incidenti del Pilastro di sicurezza del Framework AWS Well-Architected per ulteriori indicazioni su pianificazione, simulazione e risposte agli incidenti di sicurezza.

Risorse

Best practice correlate:

- SEC03-BP07 Analisi dell'accesso pubblico e multi-account
- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC01-BP07 Identificare le minacce e dare priorità alle mitigazioni utilizzando un modello di minaccia.

Documenti correlati:

- Valutazione dei metodi di controllo degli accessi per proteggere le API Amazon API Gateway
- Configurazione dell'autenticazione TLS reciproca per una REST API
- Come proteggere gli endpoint HTTP API Gateway con il sistema di autorizzazione JWT
- Autorizzazione delle chiamate dirette ai servizi AWS mediante il provider di credenziali AWS IoT Core
- AWS Security Incident Response Guide

Video correlati:

- AWS re:invent 2022: Introducing VPC Lattice
- AWS re:invent 2020: Serverless API authentication for HTTP APIs on AWS

Esempi correlati:

- Workshop Amazon VPC Lattice
- Zero-Trust Episode 1 – The Phantom Service Perimeter workshop

Risposta agli imprevisti

Domanda

- SEC 10. In che modo è possibile prevedere gli eventi, rispondere ad essi e risolverli?

SEC 10. In che modo è possibile prevedere gli eventi, rispondere ad essi e risolverli?

Anche se dispone di controlli preventivi e di rilevamento maturi, l'organizzazione deve ancora implementare meccanismi per rispondere e mitigare il potenziale impatto degli incidenti di sicurezza. La tua preparazione influisce fortemente sulla capacità dei team di operare in modo efficace durante un incidente, isolare, contenere ed eseguire indagini sui problemi e ripristinare le operazioni a uno stato valido noto. La messa in atto degli strumenti e l'accesso prima di un incidente di sicurezza, quindi la pratica sistematica della risposta agli incidenti durante le giornate di gioco, aiuterà a garantire il ripristino, riducendo al minimo le interruzioni dell'attività.

Best practice

- SEC10-BP01 Identificazione del personale chiave e delle risorse esterne
- SEC10-BP02 Sviluppo di piani di gestione degli incidenti
- SEC10-BP03 Preparazione di funzionalità forensi
- SEC10-BP04 Sviluppo e test di playbook di risposta agli incidenti di sicurezza
- SEC10-BP05 Preassegnazione dell'accesso
- SEC10-BP06 Distribuzione anticipata degli strumenti
- SEC10-BP07 Esecuzione di simulazioni
- SEC10-BP08 Definizione di un framework per apprendere dagli incidenti

SEC10-BP01 Identificazione del personale chiave e delle risorse esterne

Identifica personale, risorse e requisiti legali interni ed esterni per aiutare l'organizzazione a rispondere a un incidente.

Risultato desiderato: disporre di un elenco di persone chiave, delle loro informazioni di contatto e dei ruoli che ricoprono in caso di risposta a un evento di sicurezza. Rivedere queste informazioni regolarmente e aggiornarle per riflettere i cambiamenti del personale dal punto di vista degli strumenti interni ed esterni. Nel documentare queste informazioni si considerano tutti i fornitori di servizi e i venditori di terze parti, compresi i partner di sicurezza, i fornitori di cloud e le applicazioni software-

as-a-service (SaaS). Durante un evento di sicurezza, il personale è disponibile con il livello di responsabilità, il contesto e l'accesso appropriati per essere in grado di rispondere e recuperare.

Anti-pattern comuni:

- Non mantenere un elenco aggiornato del personale chiave con le informazioni di contatto, i ruoli e le responsabilità in caso di risposta a eventi di sicurezza.
- Dare per scontato che tutti comprendano le persone, le dipendenze, l'infrastruttura e le soluzioni per rispondere a un evento e da recuperare da un evento.
- Non disporre di un archivio di documenti o conoscenze che rappresenti l'infrastruttura o la progettazione di applicazioni chiave.
- Non disporre di processi di onboarding adeguati per i nuovi dipendenti, in modo che possano contribuire efficacemente alla risposta a un evento di sicurezza, come ad esempio la realizzazione di simulazioni di eventi.
- Non disporre di un percorso di escalation quando il personale chiave è temporaneamente non disponibile o non risponde durante gli eventi di sicurezza.

Vantaggi della definizione di questa best practice: questa pratica riduce i tempi di triage e risposta impiegati per identificare il personale giusto e i relativi ruoli durante un evento. Riduci al minimo le perdite di tempo durante un evento mantenendo un elenco aggiornato del personale chiave e dei suoi ruoli, in modo da poter portare le persone giuste al triage e al recupero da un evento.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Identifica il personale chiave all'interno della tua organizzazione: mantieni un elenco di contatti del personale all'interno dell'organizzazione che devi coinvolgere. Rivedi e aggiorna regolarmente queste informazioni in caso di spostamento del personale, come modifiche organizzative, promozioni e cambi di team. Questo è particolarmente importante per i ruoli chiave come gli incident manager, i team di risposta e i responsabili delle comunicazioni.

- Incident manager: hanno l'autorità generale durante la risposta all'evento.
- Team di risposta agli incidenti: sono responsabili delle attività di indagine e riparazione. Queste persone possono differire in base al tipo di evento, ma in genere sono sviluppatori e team operativi responsabili dell'applicazione interessata.

- Responsabile delle comunicazioni: è responsabile delle comunicazioni interne ed esterne, in particolare con gli enti pubblici, le autorità di regolamentazione e i clienti.
- Esperti in materia (SME): nel caso di team distribuiti e autonomi, ti consigliamo di identificare la figura di SME per carichi di lavoro mission critical. Queste persone offrono approfondimenti sul funzionamento e sulla classificazione dei dati dei carichi di lavoro critici coinvolti nell'evento.

Prendi in considerazione l'utilizzo della funzionalità AWS Systems Manager Incident Manager per acquisire i contatti chiave, definire un piano di risposta, automatizzare gli orari delle chiamate e creare piani di escalation. Automatizza e organizza i turni per tutto il personale attraverso un programma di chiamata, in modo che la responsabilità del carico di lavoro sia condivisa tra i proprietari. Ciò promuove buone pratiche, come l'emissione di metriche e registri pertinenti e la definizione di soglie di allarme importanti per il carico di lavoro.

Identifica i partner esterni: le aziende utilizzano strumenti realizzati da fornitori di software indipendenti (ISV), partner e subappaltatori per creare soluzioni distintive per i propri clienti. Coinvolgi il personale chiave di queste parti che può aiutarti a rispondere e a riprendersi da un incidente. Ti consigliamo di iscriverti al livello appropriato di AWS Support per ottenere un rapido accesso agli SME AWS attraverso un caso di supporto. Prendi in considerazione accordi simili con tutti i fornitori di soluzioni critiche per i carichi di lavoro. Alcuni eventi di sicurezza richiedono alle aziende quotate in borsa di notificare l'evento e gli impatti agli enti pubblici e alle autorità di regolamentazione pertinenti. Mantieni e aggiorna le informazioni di contatto per i dipartimenti pertinenti e le persone responsabili.

Passaggi dell'implementazione

1. Configura una soluzione per la gestione degli incidenti.
    a. Prendi in considerazione l'implementazione di Incident Manager nel tuo account Security Tooling.
2. Definisci i contatti nella tua soluzione di gestione degli incidenti.
    a. Definisci almeno due tipi di canali per ogni contatto (come SMS, telefono o e-mail), per garantire la raggiungibilità durante un incidente.
3. Definisci un piano di risposta.
    a. Identifica i contatti più appropriati da coinvolgere durante un incidente. Definisci piani di escalation allineati ai ruoli del personale da coinvolgere, piuttosto che ai singoli contatti. Valuta la possibilità di includere i contatti che potrebbero essere responsabili dell'informazione di entità esterne, anche se non sono direttamente coinvolti nella risoluzione dell'incidente.

Risorse

Best practice correlate:

- OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni

Documenti correlati:

- AWS Security Incident Response Guide

Esempi correlati:

- AWS customer playbook framework
- Prepare for and respond to security incidents in your AWS environment

Strumenti correlati:

- AWS Systems Manager Incident Manager

Video correlati:

- Amazon's approach to security during development

SEC10-BP02 Sviluppo di piani di gestione degli incidenti

Il primo documento da sviluppare per la risposta agli incidenti è il piano di risposta agli incidenti. Lo scopo del piano di risposta agli incidenti è costituire la base del programma e della strategia di risposta agli incidenti.

Vantaggi dell'adozione di questa best practice: Lo sviluppo di processi di risposta agli incidenti completi e chiaramente definiti è fondamentale per un programma di risposta agli incidenti efficace e scalabile. Quando si verifica un evento di sicurezza, passaggi e flussi di lavoro ben definiti ti aiuteranno a rispondere in modo tempestivo. Potrebbero essere già presenti processi di risposta agli incidenti. Indipendentemente dallo stato attuale, è importante aggiornare, iterare e testare regolarmente i processi di risposta agli incidenti.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Un piano di gestione degli incidenti è fondamentale per rispondere, mitigare e ripristinare lo stato a seguito del potenziale impatto degli incidenti di sicurezza. Un piano di gestione degli incidenti è un processo strutturato per identificare, correggere e rispondere tempestivamente agli incidenti di sicurezza.

Il cloud ha molti degli stessi ruoli e requisiti operativi che si trovano in un ambiente on-premise. Quando si crea un piano di gestione degli incidenti è importante tenere conto delle strategie di risposta e ripristino che meglio si allineano ai risultati aziendali e ai requisiti di conformità. Ad esempio, se gestisci carichi di lavoro in AWS conformi a FedRAMP negli Stati Uniti, è utile attenersi a NIST SP 800-61 Computer Security Handling Guide (NIST SP 800-61 Guida alla gestione della sicurezza informatica). Analogamente, quando gestisci carichi di lavoro con informazioni di identificazione personale (PII) europee, considera ad esempio come potresti proteggere e rispondere a problemi relativi alla residenza dei dati come richiesto dalle normative del Regolamento generale sulla protezione dei dati (GDPR) dell'Unione europea.

Quando crei un piano di gestione degli incidenti per i carichi di lavoro in AWS, inizia con il modello di responsabilità condivisa AWS per creare un approccio di difesa in profondità in risposta agli incidenti. In questo modello, AWS gestisce la sicurezza del cloud e tu sei responsabile della sicurezza nel cloud. Ciò significa che mantieni il controllo e sei responsabile dei controlli di sicurezza che scegli di implementare. La AWS Security Incident Response Guide (Guida alle risposte agli incidenti di sicurezza di AWS) illustra i concetti chiave e le linee guida di base per la creazione di un piano di gestione degli incidenti incentrato sul cloud.

Un piano di gestione degli incidenti efficace deve essere continuamente iterato per rimanere in linea con l'obiettivo delle operazioni cloud. Prendi in considerazione l'utilizzo dei piani di implementazione descritti di seguito durante la creazione e l'evoluzione del tuo piano di gestione degli incidenti.

Passaggi dell'implementazione

Definizione di ruoli e responsabilità

La gestione degli eventi di sicurezza richiede disciplina interorganizzativa e propensione all'azione. All'interno della struttura organizzativa, dovrebbero esserci molte persone da considerarsi responsabili, affidabili, consultabili o informate durante un incidente, come i rappresentanti delle risorse umane (HR), i membri del team esecutivo e quelli dell'ufficio legale. Considera questi ruoli e queste responsabilità e se è necessario coinvolgere terze parti. Si noti che molte aree geografiche hanno leggi locali che regolano cosa dovrebbe e non dovrebbe essere fatto. Sebbene possa

sembrare burocratico creare una tabella delle persone responsabili, affidabili, consultabili e informate (RACI) per i piani di risposta relativi alla sicurezza, ciò facilita una comunicazione rapida e diretta e delinea chiaramente la leadership nelle diverse fasi dell'evento.

Durante un incidente, includere i proprietari e gli sviluppatori delle applicazioni e delle risorse interessate è fondamentale perché sono esperti in materia (PMI) che possono fornire informazioni e contesto per aiutare a valutare l'impatto. Assicurati di fare pratica e instaurare relazioni con gli sviluppatori e i proprietari delle applicazioni prima di affidarti alla loro esperienza per la gestione della risposta agli incidenti. I proprietari di applicazioni o le PMI, come gli amministratori o gli ingegneri del cloud, potrebbero dover intervenire in situazioni in cui l'ambiente non è noto oppure è complesso o chi risponde non ha accesso all'ambiente interessato.

Infine, nell'indagine o nella risposta potrebbero essere coinvolti partner affidabili perché possono fornire competenze aggiuntive e capacità analitiche strategiche. Quando non disponi di queste competenze nel tuo team, potresti voler assumere una persona esterna per assistenza.

Analisi del team di risposta di AWS e del supporto

- AWS Support
  - AWS Support offre un'ampia gamma di piani che forniscono accesso agli strumenti e alla competenza che genera successo e stato operativo delle soluzioni AWS. Se hai bisogno di supporto tecnico e di ulteriori risorse per pianificare, implementare e ottimizzare il tuo ambiente AWS, puoi selezionare il piano di supporto più adatto al tuo caso d'uso AWS.
  - Valuta l'ipotesi di utilizzare il Centro di supporto in AWS Management Console (è richiesto l'accesso) come punto di contatto centralizzato per ottenere assistenza per problemi che riguardano le tue risorse AWS. L'accesso a AWS Support è controllato da AWS Identity and Access Management. Per ulteriori informazioni sull'accesso alle funzionalità AWS Support, consulta la sezione Nozioni di base su AWS Support.
- Team di risposta agli incidenti dei clienti AWS (CIRT)
  - Il Team di risposta agli incidenti dei clienti AWS (CIRT) è un team AWS globale specializzato disponibile 24 ore su 24, 7 giorni su 7, che fornisce supporto ai clienti durante eventi di sicurezza attivi sul lato cliente del modello di responsabilità condivisa AWS.
  - Quando il team AWS CIRT ti supporta, fornisce assistenza nella valutazione e nel ripristino di un evento di sicurezza attivo AWS. Può aiutare nell'analisi delle cause principali con l'uso dei log dei servizi AWS e fornire suggerimenti per il ripristino. Può anche fornire consigli e best practice sulla sicurezza per aiutarti a evitare eventi di sicurezza in futuro.
  - I clienti AWS possono coinvolgere il team AWS CIRT attraverso un caso AWS Support.

- Supporto per la risposta agli attacchi DDoS
  - AWS offre AWS Shield, che fornisce un servizio di protezione DDoS (Distributed Denial of Service) gestito che protegge le applicazioni Web in esecuzione su AWS. Shield fornisce un rilevamento sempre attivo e mitigazioni automatiche in linea che possono ridurre al minimo i tempi di inattività e la latenza delle applicazioni, quindi non è necessario utilizzare AWS Support per avvalersi della protezione dagli attacchi DDoS. Esistono due livelli di Shield: AWS Shield Standard e AWS Shield Advanced. Per maggiori informazioni sulle differenze tra questi due livelli, consulta la documentazione delle funzionalità di Shield.
- AWS Managed Services (AMS)
  - AWS Managed Services (AMS) offre gestione continua dell'infrastruttura AWS, così potrai occuparti a tempo pieno delle tue applicazioni. Grazie all'implementazione delle best practice per la gestione dell'infrastruttura, AMS riduce il sovraccarico operativo e il livello di rischio. AMS automatizza attività frequenti quali richieste di modifica, monitoraggio, gestione di patch, sicurezza e backup, nonché fornisce servizi completi per il ciclo di vita per gestire provisioning, esecuzione e supporto dell'infrastruttura.
  - AMS è responsabile dell'implementazione di una suite di controlli di sicurezza e fornisce una risposta di prima linea agli avvisi 24 ore su 24, 7 giorni su 7. Quando viene avviato un avviso, AMS segue una serie standard di playbook automatici e manuali per verificare una risposta coerente. Questi playbook vengono condivisi con i clienti AMS durante l'onboarding in modo che possano sviluppare e coordinare una risposta con AMS.

Sviluppo di piani di risposta agli incidenti

Lo scopo del piano di risposta agli incidenti è costituire la base del programma e della strategia di risposta agli incidenti. Il piano di risposta agli incidenti deve essere contenuto in un documento formale. Un piano di risposta agli incidenti include in genere le seguenti sezioni:

- Una panoramica del team di risposta agli incidenti: delinea gli obiettivi e le funzioni del team di risposta agli incidenti.
- Ruoli e responsabilità: elenca le parti interessate alla risposta agli incidenti e descrive in dettaglio i loro ruoli quando si verifica un incidente.
- Un piano di comunicazione: dettagli sulle informazioni di contatto e su come comunichi durante un incidente.
- Metodi di comunicazione di backup: è consigliabile utilizzare la comunicazione fuori banda come backup in caso di incidente. Un esempio di applicazione che fornisce un canale di comunicazione fuori banda sicuro è AWS Wickr.

- Fasi di risposta agli incidenti e azioni da intraprendere: enumera le fasi della risposta agli incidenti (ad esempio, rilevamento, analisi, eliminazione, contenimento e ripristino), comprese le azioni di alto livello da intraprendere all'interno di tali fasi.
- Definizioni di gravità e prioritizzazione degli incidenti: descrive in dettaglio come classificare la gravità di un incidente, come assegnare la priorità all'incidente e, quindi, in che modo le definizioni di gravità influiscono sulle procedure di escalation.

Sebbene queste sezioni siano comuni a società di diverse dimensioni e settori, il piano di risposta agli incidenti di ciascuna organizzazione è unico. Devi creare un piano di risposta agli incidenti che funzioni al meglio per la tua organizzazione.

Risorse

Best practice correlate:

- SEC 4 (In che modo individui ed esamini gli eventi di sicurezza?)

Documenti correlati:

- AWS Security Incident Response Guide (Guida alle risposte agli incidenti di sicurezza di AWS)
- NIST: Guida alla gestione degli incidenti di sicurezza informatica

SEC10-BP03 Preparazione di funzionalità forensi

Prima che si verifichi un incidente di sicurezza, puoi sviluppare le funzionalità forensi per supportare le indagini sugli eventi di sicurezza.

Livello di rischio associato se questa best practice non fosse adottata: medio

Il concetto della tradizionale analisi forense on-premise si applica ad AWS. Per informazioni chiave su come iniziare a sviluppare funzionalità forensi in Cloud AWS, consulta Forensic investigation environment strategies in the Cloud AWS.

Una volta configurati l'ambiente e la struttura di Account AWS per le funzionalità forensi, definisci le tecnologie necessarie in modo da eseguire efficacemente le metodologie forensi in quattro fasi:

- Raccolta: acquisisci i log AWS pertinenti, come quelli di AWS CloudTrail, AWS Config, del flusso VPC e dell'host. Raccogli snapshot, backup e dump di memoria delle risorse AWS interessate, se disponibili.

- Esame: rivedi i dati raccolti estraendo e valutando le informazioni pertinenti.
- Analisi: studia i dati raccolti per comprendere l'incidente e trarre le conclusioni.
- Segnalazione: presenta le informazioni risultanti dalla fase di analisi.

Passaggi dell'implementazione

Preparazione dell'ambiente per le funzionalità forensi

AWS Organizations ti aiuta a gestire e governare centralmente un ambiente AWS mentre le risorse AWS crescono e si dimensionano. Un'organizzazione AWS consolida gli Account AWS in modo da poterli amministrare come una singola unità. È possibile utilizzare le unità organizzative per raggruppare gli account e amministrarli come singola unità.

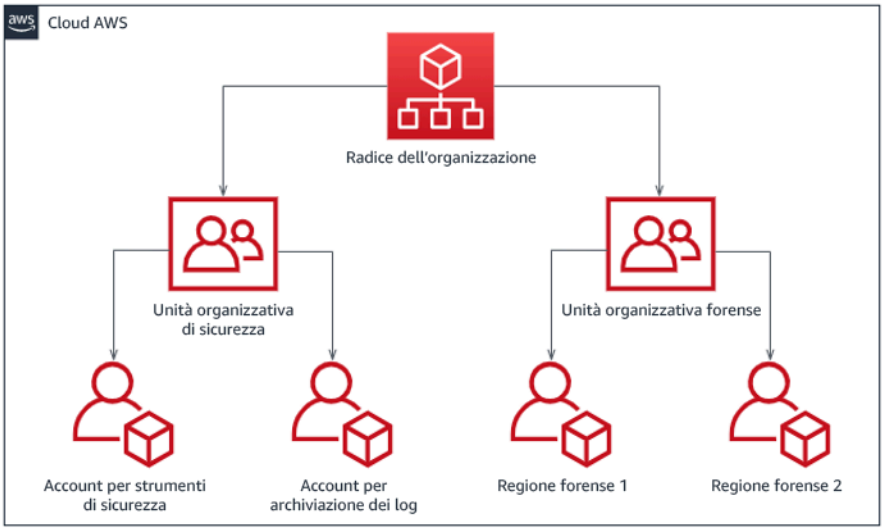
Per rispondere agli incidenti è utile disporre di una struttura di Account AWS che supporti le funzioni di risposta agli incidenti e includa una Unità organizzativa di sicurezza e una Unità organizzativa forense. All'interno dell'unità organizzativa di sicurezza, è necessario disporre degli account per:

- Archiviazione dei log: aggrega i log in un Account AWS di archiviazione dei log con autorizzazioni limitate.
- Strumenti di sicurezza: centralizza i servizi di sicurezza in un Account AWS dello strumento di sicurezza. Questo account funge da amministratore delegato per i servizi di sicurezza.

Nell'unità organizzativa di funzionalità forensi, hai la possibilità di implementare uno o più account di funzionalità forensi per ogni regione in cui operi, a seconda di quale è più adatta all'azienda e al modello operativo. Se crei un account di funzionalità forensi per regione, puoi bloccare la creazione di risorse AWS al di fuori della regione e ridurre il rischio che le risorse vengano copiate in una regione indesiderata. Ad esempio, se operi solo in US East (N. Virginia) Region (`us-east-1`) e US West (Oregon) (`us-west-2`), allora avresti due account nell'unità organizzativa forense: uno per `us-east-1` e uno per `us-west-2`.

Puoi creare un Account AWS di funzionalità forensi per più regioni. Quando si copiano le risorse AWS nell'account occorre prestare attenzione a rispettare i requisiti di sovranità dei dati. Poiché la creazione di nuovi account richiede tempo, è fondamentale creare e strumentare gli account di funzionalità forensi con largo anticipo rispetto agli incidenti, in modo che gli addetti siano preparati a utilizzarli efficacemente per la risposta.

Il diagramma seguente mostra una struttura degli account di esempio che include un'unità organizzativa di funzionalità forensi con gli account di funzionalità forensi per regione:



Struttura degli account per regione per la risposta agli incidenti

Acquisizione di backup e snapshot

La configurazione dei backup dei sistemi e dei database importanti è fondamentale per il ripristino da un incidente di sicurezza e per scopi forensi. Con i backup puoi ripristinare i tuoi sistemi allo stato di sicurezza precedente. In AWS puoi acquisire snapshot di varie risorse. Gli snapshot forniscono i backup point-in-time delle risorse. Esistono molti servizi AWS che possono supportarti nelle operazioni di backup e ripristino. Per informazioni dettagliate su questi servizi e approcci per il backup e il ripristino, consultare Guida prescrittiva per il backup e il ripristino e Usa i backup per il ripristino in seguito a incidenti di sicurezza.

Soprattutto in situazioni come un attacco ransomware, è fondamentale che i backup siano ben protetti. Per indicazioni sulla protezione dei backup, consultare Le 10 migliori pratiche di sicurezza per proteggere i backup in AWS. Oltre a proteggere, è necessario eseguire regolarmente i test dei processi di backup e ripristino per verificare che la tecnologia e le procedure in uso funzionino come previsto.

Automazione delle funzionalità forensi

Durante un evento di sicurezza, il team addetto a rispondere agli incidenti deve essere in grado di raccogliere e analizzare rapidamente le prove, mantenendo la precisione per il periodo di tempo relativo all'evento (ad esempio acquisendo i log relativi a una risorsa o un evento specifico o raccogliendo il dump della memoria di un'istanza Amazon EC2). Per il team addetto a rispondere agli incidenti è difficile e dispendioso in termini di tempo raccogliere manualmente le prove pertinenti, soprattutto se le istanze e gli account sono numerosi. Inoltre, la raccolta manuale può essere

soggetta all'errore umano. Per questi motivi, è necessario sviluppare e implementare il più possibile l'automazione per le funzionalità forensi.

AWS offre una serie di risorse di automazione per le funzionalità forensi, elencate nella sezione Risorse di seguito. Queste risorse sono esempi di modelli di funzionalità forensi che abbiamo sviluppato e che i clienti hanno implementato. Costituiscono un'utile architettura di riferimento per iniziare, ma prendi in considerazione la possibilità di modificarli o creare nuovi modelli di automazione per le funzionalità forensi in base all'ambiente, ai requisiti, agli strumenti e ai processi forensi.

Risorse

Documenti correlati:

- AWS Security Incident Response Guide - Develop Forensics Capabilities
- AWS Security Incident Response Guide - Forensics Resources
- Forensic investigation environment strategies in the Cloud AWS
- How to automate forensic disk collection in AWS
- AWS Prescriptive Guidance - Automate incident response and forensics

Video correlati:

- Automatizzazione delle indagini e della risposta agli incidenti

Esempi correlati:

- Automated Incident Response and Forensics Framework
- Automated Forensics Orchestrator for Amazon EC2

SEC10-BP04 Sviluppo e test di playbook di risposta agli incidenti di sicurezza

Una parte fondamentale della preparazione dei processi di risposta agli incidenti è costituita dallo sviluppo di playbook. I playbook di risposta agli incidenti forniscono una serie di indicazioni prescrittive e di passaggi da seguire quando si verifica un evento di sicurezza. Avere una struttura e passaggi chiari semplifica la risposta e riduce la probabilità di errore umano.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

È necessario creare i playbook per scenari di incidenti come:

- Incidenti previsti: i playbook devono essere creati per gli incidenti previsti. tra cui minacce come il Denial of Service (DoS), il ransomware e la compromissione delle credenziali.
- Avvisi o esiti di sicurezza noti: i playbook devono essere creati per gli esiti e gli avvisi di sicurezza noti, ad esempio gli esiti GuardDuty. Potresti ricevere un risultato di GuardDuty e non sapere cosa fare. Per evitare di mal gestire o ignorare un risultato di GuardDuty, crea un playbook per ogni potenziale risultato di GuardDuty. I dettagli e le indicazioni sulla correzione sono disponibili nella documentazione di GuardDuty. Vale la pena notare che GuardDuty non è abilitato per impostazione predefinita e comporta costi. Per maggiori dettagli su GuardDuty, consulta Appendice A: Definizioni delle capacità del cloud - Visibilità e avvisi.

I playbook devono contenere i passaggi tecnici che un analista deve completare per indagare e rispondere adeguatamente a un potenziale incidente di sicurezza.

Passaggi dell'implementazione

Gli elementi da includere in un playbook sono:

- Panoramica del playbook: quale scenario di rischio o incidente affronta questo playbook? Qual è l'obiettivo del playbook?
- Prerequisiti: quali log, meccanismi di rilevamento e strumenti automatizzati sono necessari per questo scenario di incidente? Qual è la notifica prevista?
- Informazioni sulla comunicazione e sull'escalation: chi è coinvolto e quali sono le loro informazioni di contatto? Quali sono le responsabilità di ogni stakeholder?
- Fasi di risposta: in tutti i passaggi per la risposta agli incidenti, quali misure tattiche devono essere prese? Quali query deve eseguire l'analista? Quale codice deve essere eseguito per ottenere il risultato desiderato?
  - Individuazione: come verrà rilevato l'incidente?
  - Analisi: come verrà determinato l'ambito dell'impatto?
  - Contenimento: come verrà isolato l'incidente per limitarne la portata?
  - Sradicamento: come verrà rimossa la minaccia dall'ambiente?
  - Recupero: in che modo il sistema o la risorsa interessati verranno riportati in produzione?
- Risultati attesi: dopo l'esecuzione delle query e del codice, qual è il risultato previsto del playbook?

Risorse

Best practice Well-Architected correlate:

- SEC10-BP02 Sviluppo di piani di gestione degli incidenti

Documenti correlati:

- Framework for Incident Response Playbooks
- Develop your own Incident Response Playbooks
- Incident Response Playbook Samples
- Building an AWS incident response runbook using Jupyter playbooks and CloudTrail Lake

SEC10-BP05 Preassegnazione dell'accesso

Verifica che il team di risposta agli incidenti disponga degli opportuni diritti di accesso allocati in AWS per ridurre i tempi necessari per l'analisi e il ripristino.

Anti-pattern comuni:

- L'utilizzo dell'account root per la risposta agli incidenti.
- La modifica degli account utente esistenti.
- La manipolazione diretta delle autorizzazioni IAM quando si fornisce l'elevazione dei privilegi just-in-time.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

AWS raccomanda di ridurre o eliminare, ove possibile, la dipendenza da credenziali di lunga durata, a favore delle credenziali temporanee e dei meccanismi di escalation dei privilegi just-in-time . Le credenziali di lunga durata sono soggette a rischi per la sicurezza e aumentano il sovraccarico operativo. Per la maggior parte delle attività di gestione, nonché per le attività di risposta agli incidenti, consigliamo di implementare la federazione delle identità insieme all'escalation temporanea per l'accesso amministrativo. In questo modello, un utente richiede l'elevazione a un livello di privilegio superiore (come un ruolo di risposta agli incidenti) e, se è idoneo all'elevazione, la richiesta viene inviata al responsabile dell'approvazione. Se la richiesta viene approvata, l'utente riceve un

set di credenziali AWS temporanee che può utilizzare per eseguire le sue attività. Alla scadenza di queste credenziali, l'utente deve inviare una nuova richiesta di elevazione.

Si consiglia l'uso dell'escalation temporanea dei privilegi nella maggior parte degli scenari di risposta agli incidenti. Il modo corretto per farlo è utilizzare AWS Security Token Service e le policy di sessione per definire l'ambito di accesso.

Esistono scenari in cui le identità federate non sono disponibili, come nei casi di:

- Interruzione correlata a un gestore dell'identità digitale (IdP) compromesso.
- Configurazione errata o errore umano che causa l'interruzione del sistema di gestione dell'accesso federato.
- Attività dannose come un evento DDoS (Distributed Denial of Service) o indisponibilità del sistema.

Nei casi precedenti, si deve configurare un accesso di emergenza di tipo break-glass per consentire l'analisi e la tempestiva risoluzione degli incidenti. Ti consigliamo di utilizzare un utente IAM con le autorizzazioni appropriate per eseguire le attività e accedere alle risorse AWS. Utilizza le credenziali root solo per le attività che richiedono l'accesso come utente root. Per verificare che i team di risposta agli incidenti dispongano del corretto livello di accesso ad AWS e ad altri sistemi pertinenti, ti consigliamo di eseguire la pre-assegnazione di account utente dedicati. Gli account utente richiedono l'accesso con privilegi e devono essere rigorosamente controllati e monitorati. Gli account devono essere creati con il minor numero di privilegi richiesti per eseguire le attività e il livello di accesso deve essere basato sui playbook inclusi nel piano di gestione degli incidenti.

Utilizza utenti e ruoli specifici e dedicati come best practice. L'escalation temporanea dell'accesso di utenti o ruoli tramite l'aggiunta di policy IAM rende poco chiaro quale fosse l'accesso degli utenti durante l'incidente e rischia di non revocare i privilegi oggetto di escalation.

È importante rimuovere il maggior numero possibile di dipendenze per verificare che sia possibile ottenere l'accesso nel maggior numero possibile di scenari di errore. Per supportare questa esigenza, crea un playbook per verificare che gli utenti dei team di risposta agli incidenti vengano creati come utenti AWS Identity and Access Management in un account di sicurezza dedicato e non gestiti tramite una federazione esistente o una soluzione di autenticazione unica (SSO). Ogni singolo utente dei team di risposta deve avere il proprio account denominato. La configurazione dell'account deve applicare una policy di password complesse e l'autenticazione a più fattori (MFA). Se i playbook di risposta agli incidenti richiedono solo l'accesso alla AWS Management Console, non è necessario che l'utente disponga di chiavi di accesso configurate né che sia esplicitamente autorizzato a creare chiavi di accesso. A tale scopo è possibile configurare le policy IAM o le policy di controllo dei servizi

come menzionato in AWS Security Best Practices (Best practice di sicurezza AWS) per le policy di controllo dei servizi AWS Organizations. Gli utenti non devono avere privilegi oltre la capacità di assumere i ruoli di risposta agli incidenti in altri account.

Durante un incidente potrebbe essere necessario concedere l'accesso ad altre persone interne o esterne per supportare le attività di analisi, correzione o ripristino. In questo caso, utilizza il meccanismo del playbook menzionato in precedenza e un processo per verificare che qualsiasi accesso aggiuntivo venga revocato immediatamente dopo il completamento dell'incidente.

Per verificare che l'uso dei ruoli di risposta agli incidenti possa essere adeguatamente monitorato e controllato, è essenziale che gli account utente IAM creati a tale scopo non siano condivisi tra le persone e che l'utente root Account AWS non venga utilizzato se non per un'attività specifica. Se è richiesto l'utente root (ad esempio, l'accesso IAM a un account specifico non è disponibile), utilizza un processo separato con un playbook disponibile per verificare la disponibilità della password dell'utente root e del token MFA.

Per configurare le policy IAM per i ruoli di risposta agli incidenti, prendi in considerazione di usare IAM Access Analyzer per generare le policy sulla base dei log AWS CloudTrail. In questo caso, concedi l'accesso come amministratore al ruolo di risposta agli incidenti per un account non di produzione ed esegui i playbook. Al termine, potrà essere creata una policy che consenta solo le azioni da intraprendere. Questa policy può quindi essere applicata a tutti i ruoli di risposta agli incidenti in tutti gli account. Puoi anche creare una policy IAM separata per ogni playbook per avere una gestione e un controllo più semplici. Esempi di playbook possono essere piani di risposta per ransomware, violazioni dei dati, perdita dell'accesso alla produzione e altri scenari.

Utilizza gli account utente di risposta agli incidenti per assumere i ruoli di risposta IAM dedicati in altri Account AWS. Questi ruoli devono essere configurati in modo che possano essere assunti solo dagli utenti nell'account di sicurezza e la relazione di trust deve richiedere che il principale chiamante sia autenticato tramite MFA. I ruoli devono utilizzare policy IAM con ambito limitato per controllare l'accesso. Assicurati che tutte le richieste `AssumeRole` per questi ruoli vengano registrate in CloudTrail e notificate e che tutte le azioni intraprese utilizzando questi ruoli vengano registrate.

Ti consigliamo vivamente di nominare chiaramente gli account utente IAM e i ruoli IAM per trovarli facilmente nei log CloudTrail. Un esempio potrebbe essere quello di nominare gli account IAM *<ID_UTENTE>*`-BREAK-GLASS` e i ruoli IAM `RUOLO-BREAK-GLASS`.

CloudTrail viene utilizzato per registrare l'attività API negli account AWS e deve essere utilizzato per configurare gli avvisi sull'utilizzo dei ruoli di risposta agli incidenti. Fai riferimento al post del blog sulla configurazione degli avvisi quando vengono utilizzate le chiavi root. Le istruzioni possono essere

modificate per configurare il parametro Amazon CloudWatch da filtro a filtro negli eventi `AssumeRole` correlati al ruolo IAM di risposta agli incidenti:

```
{ $.eventName = "AssumeRole" && $.requestParameters.roleArn =
 "<ARN_RUOLO_DI_RISPOSTA_AGLI_INCIDENTI>" && $.userIdentity.invokedBy NOT EXISTS &&
 $.eventType != "AwsServiceEvent" }
```

Poiché è probabile che i ruoli di risposta agli incidenti abbiano un livello di accesso elevato, è importante che questi avvisi vengano inviati a un gruppo ampio e vengano gestiti tempestivamente.

Durante un incidente, è possibile che un membro del team di risposta richieda l'accesso a sistemi che non sono direttamente protetti da IAM, ad esempio istanze Amazon Elastic Compute Cloud, database Amazon Relational Database Service o piattaforme Software-as-a-service (SaaS). Anziché i protocolli nativi come SSH o RDP, ti consigliamo vivamente di utilizzare AWS Systems Manager Session Manager per l'accesso amministrativo completo alle istanze Amazon EC2. Questo accesso può essere monitorato utilizzando IAM, che è sicuro e controllato. Puoi anche automatizzare parti dei tuoi playbook utilizzando i documenti di AWS Systems Manager Run Commandche possono ridurre gli errori dell'utente e migliorare i tempi di ripristino. Per l'accesso a database e strumenti di terze parti, ti consigliamo di archiviare le credenziali di accesso in AWS Secrets Manager e di concedere l'accesso ai ruoli degli utenti dei team di risposta agli incidenti.

Infine, la gestione degli account utente IAM di risposta agli incidenti deve essere aggiunta ai processi degli utenti che si uniscono, si spostano o lasciano l'organizzazione e deve rivista e testata periodicamente per verificare che sia consentito solo l'accesso previsto.

Risorse

Documenti correlati:

- Managing temporary elevated access to your AWS environment (Gestione dell'accesso temporaneo con privilegi elevati all'ambiente AWS)
- AWS Security Incident Response Guide (Guida alle risposte agli incidenti di sicurezza di AWS)
- AWS Elastic Disaster Recovery
- AWS Systems Manager Incident Manager
- Impostazione di una policy delle password dell'account per utenti IAM
- Utilizzo dell'autenticazione a più fattori (MFA) in AWS
- Configuring Cross-Account Access with MFA (Configurazione dell'accesso multi-account con MFA)

- Using IAM Access Analyzer to generate IAM policies (Utilizzo di IAM Access Analyzer per generare policy IAM)
- Best Practices for AWS Organizations Service Control Policies in a Multi-Account Environment (Best practice per le policy di controllo dei servizi di AWS Organizations in un ambiente multi-account)
- How to Receive Notifications When Your AWS Account's Root Access Keys Are Used (Come ricevere le notifiche quando vengono utilizzate le chiavi di accesso root dell'account AWS)
- Create fine-grained session permissions using IAM managed policies (Creazione di autorizzazioni di sessione dettagliate utilizzando le policy gestite da IAM)

Video correlati:

- Automating Incident Response and ForensicsAWS
- DIY guide to runbooks, incident reports, and incident response
- Prepare for and respond to security incidents in your AWS environment

Esempi correlati:

- Lab: AWS Account Setup and Root User (Laboratorio: configurazione dell'account AWS e dell'utente root)
- Lab: Incident Response with AWS Console and CLI (Laboratorio: risposta agli incidenti con la Console AWS e l'interfaccia della riga di comando)

SEC10-BP06 Distribuzione anticipata degli strumenti

Verifica che il team addetto alla sicurezza disponga degli strumenti giusti pre-distribuiti per ridurre i tempi di indagine fino al ripristino.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Per automatizzare le funzioni delle operazioni e la risposta di sicurezza, puoi utilizzare un set completo di API e strumenti AWS. Puoi automatizzare completamente le funzionalità di gestione delle identità, sicurezza della rete, protezione dei dati e monitoraggio e distribuirle utilizzando metodi di sviluppo software comuni già esistenti. Quando crei l'automazione della sicurezza, il sistema può

monitorare, rivedere e avviare una risposta, invece di far monitorare alle persone il comportamento di sicurezza e reagire manualmente agli eventi.

Se i team di risposta agli incidenti continuano a rispondere agli avvisi nello stesso modo, rischiano il cosiddetto affaticamento dagli avvisi ("alert fatigue"). Ciò significa che, nel corso del tempo, il team può diventare desensibilizzato agli avvisi e può commettere errori nella gestione di situazioni ordinarie o farsi sfuggire avvisi insoliti. L'automazione aiuta a evitare l'affaticamento dagli avvisi utilizzando funzioni che elaborano gli avvisi ripetitivi e ordinari, lasciando alle persone la gestione degli incidenti sensibili e univoci. Se si integrano sistemi di rilevamento delle anomalie, come Amazon GuardDuty, AWS CloudTrail Insights e Amazon CloudWatch Anomaly Detection, è possibile ridurre l'impatto di avvisi frequenti basati su soglie.

Puoi migliorare i processi manuali automatizzando le fasi del processo a livello di programmazione. Dopo aver definito il modello di correzione di un evento, puoi decomporre tale modello in una logica fruibile e scrivere il codice per eseguire tale logica. Il team addetto alla risposta può quindi eseguire il codice per risolvere il problema. Nel corso del tempo, puoi automatizzare più fasi e, infine, gestire automaticamente intere classi di incidenti comuni.

Durante un'indagine di sicurezza, devi essere in grado di esaminare i log pertinenti per registrare e comprendere l'intera portata e la tempistica dell'incidente. I log sono necessari anche per la generazione di avvisi, che indicano che sono avvenute determinate azioni di interesse. È fondamentale selezionare, attivare, memorizzare e impostare i meccanismi di query e recupero e impostare gli avvisi. Inoltre, un modo efficace per fornire gli strumenti per la ricerca nei dati di log è Amazon Detective.

AWS offre oltre 200 servizi cloud e migliaia di funzionalità. Ti consigliamo di esaminare i servizi che possono supportare e semplificare la tua strategia di risposta agli incidenti.

Oltre ai log, è necessario sviluppare e implementare una strategia di applicazione dei tag. L'applicazione dei tag può aiutare a fornire il contesto per lo scopo di una risorsa AWS. I tag può essere utilizzati anche per l'automazione.

Passaggi dell'implementazione

Seleziona e configura i log per l'analisi e gli avvisi

Consulta la seguente documentazione sulla configurazione dei log per la risposta agli incidenti:

- Logging strategies for security incident response
- SEC04-BP01 Configurazione dei registri di servizi e applicazioni

Abilitazione dei servizi di sicurezza per supportare il rilevamento e la risposta

AWS offre funzionalità investigative, preventive e reattive native e altri servizi che possono essere utilizzati per progettare soluzioni di sicurezza personalizzate. Per un elenco dei servizi più pertinenti per la risposta agli incidenti di sicurezza, consulta Definizioni delle capacità del cloud.

Sviluppa e implementa una strategia di tag

Ottenere informazioni contestuali sul caso d'uso aziendale e sugli stakeholder interni pertinenti relativi a una risorsa AWS può essere difficile. Un modo per farlo è rappresentato dai tag che assegnano i metadati alle risorse AWS e sono composti da una chiave e un valore definiti dall'utente. Puoi creare i tag per classificare le risorse per scopo, proprietario, ambiente, tipo di dati elaborati e altri criteri di tua scelta.

Avere una strategia di tag coerente può accelerare le risposta e ridurre al minimo il tempo dedicato al contesto organizzativo, consentendo di identificare e discernere rapidamente le informazioni contestuali su una risorsa AWS. I tag possono anche fungere da meccanismo per avviare le automazioni di risposta. Per maggiori dettagli su cosa etichettare, consulta Etichettare le tue risorse AWS. Dovrai prima definire i tag nella tua organizzazione e quindi implementare e applicare questa strategia di tag. Per maggiori dettagli sull'implementazione e l'applicazione, consulta Implementa una strategia di etichettatura delle risorse AWS utilizzando AWS Tag Policies and Service Control Policies (SCPs).

Risorse

Best practice Well-Architected correlate:

- SEC04-BP01 Configurazione dei registri di servizi e applicazioni
- SEC04-BP02 Acquisizione di log, esiti e metriche in posizioni standardizzate.

Documenti correlati:

- Logging strategies for security incident response
- Incident response cloud capability definitions

Esempi correlati:

- Threat Detection and Response with Amazon GuardDuty and Amazon Detective
- Security Hub Workshop

- Vulnerability Management with Amazon Inspector

SEC10-BP07 Esecuzione di simulazioni

Man mano che le organizzazioni crescono e si evolvono nel tempo, aumentano anche le tipologie di minacce. Per questo motivo, è importante rivedere continuamente le capacità di risposta agli incidenti. L'esecuzione di simulazioni (note anche come giornate di gioco) è un metodo che può essere utilizzato per eseguire questa valutazione. Le simulazioni utilizzano scenari di eventi di sicurezza reali progettati per simulare le tattiche, le tecniche e le procedure (TTP) di un autore di minacce e consentire a un'organizzazione di esercitarsi e valutare le proprie capacità di risposta agli incidenti rispondendo a questi finti eventi informatici così come potrebbero verificarsi nella realtà.

Vantaggi dell'adozione di questa best practice: le simulazioni offrono una serie di vantaggi:

- Convalida della preparazione informatica e sviluppo della fiducia dei team di risposta agli incidenti.
- Verifica della precisione e dell'efficienza di strumenti e flussi di lavoro.
- Perfezionamento dei metodi di comunicazione ed escalation in linea con il piano di risposta agli incidenti.
- Opportunità di rispondere per i vettori meno comuni.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Esistono tre tipi principali di simulazioni:

- Simulazioni di situazioni di emergenza: le simulazioni di situazioni di emergenza sono sessioni basate sulla discussione che coinvolgono le varie parti interessate alla risposta agli incidenti per mettere in pratica ruoli e responsabilità e utilizzare strumenti e playbook di comunicazione consolidati. Lo svolgimento dell'esercitazione può in genere essere eseguito in un'intera giornata in un luogo virtuale, in un luogo fisico o in una combinazione di questi tipi di luogo. Poiché è basato sulla discussione, questo tipo di esercitazione si concentra su processi, persone e collaborazione. La tecnologia è parte integrante della discussione, ma l'uso effettivo di strumenti o script di risposta agli incidenti in genere non rientra in questo tipo di simulazione.
- Esercizi del team viola: questo tipo di esercitazioni aumenta il livello di collaborazione tra i team di risposta agli incidenti (team blu) e gli attori delle minacce simulate (team rosso). Il team blu è composto da membri del Security Operations Center (SOC), ma può includere anche altre parti interessate che sarebbero coinvolte durante un vero e proprio evento informatico. Il team rosso

è composto da un team responsabile dei test di penetrazione (pen-test) o da parti interessate chiave esperte in materia di sicurezza informatica. Il team rosso lavora assieme ai coordinatori dell'esercitazione durante la progettazione di uno scenario in modo che lo scenario sia accurato e fattibile. Durante le esercitazioni del team viola, l'attenzione è rivolta principalmente ai meccanismi di rilevamento, agli strumenti e alle procedure operative standard (SOP) a supporto della risposta agli incidenti.

- Esercizi del team rosso: durante un'esercitazione con il team rosso, l'attacco (team rosso) effettua una simulazione per raggiungere un determinato obiettivo o una serie di obiettivi da un ambito predeterminato. I difensori (team blu) non saranno necessariamente a conoscenza della portata e della durata dell'esercitazione, il che fornisce una valutazione più realistica di come risponderebbero a un incidente reale. Poiché le esercitazioni del team rosso possono basarsi su test invasivi, sii cauto e implementa controlli per verificare che l'esercitazione non causi danni effettivi all'ambiente.

Prendi in considerazione la possibilità di svolgere simulazioni informatiche a intervalli regolari. Ogni tipo di esercitazione può offrire vantaggi unici ai partecipanti e all'organizzazione nel suo insieme; potresti, quindi, scegliere di iniziare con tipi di simulazione meno complessi (come le simulazioni di situazioni di emergenza) e passare a tipi di simulazione più complessi (esercitazioni del team rosso). È necessario selezionare un tipo di simulazione in base alla maturità, alle risorse e ai risultati desiderati a livello di sicurezza. Alcuni clienti potrebbero scegliere di non eseguire le esercitazioni del team rosso a causa della loro complessità e dei loro costi.

Passaggi dell'implementazione

Indipendentemente dal tipo di simulazione scelto, le simulazioni sono in genere caratterizzate dai seguenti passaggi di implementazione:

1. Definisci gli elementi principali dell'esercizio: definisci lo scenario di simulazione e gli obiettivi della simulazione. Lo scenario e gli obiettivi dovrebbero essere entrambi accettati dalla leadership.
2. Identifica le principali parti interessate: come minimo, un esercizio richiede la presenza di facilitatori e partecipanti. A seconda dello scenario, potrebbero essere coinvolte altre parti interessate come la leadership legale, delle comunicazioni o esecutiva.
3. Crea ed esegui il test dello scenario: potrebbe essere necessario ridefinire lo scenario durante la creazione se risulta impossibile implementare elementi specifici. Come risultato di questa fase è previsto uno scenario definitivo.
4. Facilita la simulazione: il tipo di simulazione determina il tipo di svolgimento usato (uno scenario basato su supporto cartaceo o uno scenario con simulazione altamente tecnologica). I coordinatori

dovrebbero allineare le loro tattiche di svolgimento agli oggetti dell'esercitazione e dovrebbero coinvolgere tutti i partecipanti ove possibile per ottimizzare i benefici.

5. Sviluppa il report post-azione (AAR): individua le aree che sono andate bene, quelle che possono essere migliorate e le potenziali lacune. Il report AAR dovrebbe misurare l'efficacia della simulazione e la risposta del team all'evento simulato in modo che i progressi possano essere monitorati nel tempo con simulazioni future.

Risorse

Documenti correlati:

- AWS Incident Response Guide

Video correlati:

- AWS GameDay - Security Edition

SEC10-BP08 Definizione di un framework per apprendere dagli incidenti

L'implementazione di un framework basato sulle lezioni apprese e di una capacità di analisi delle cause principali non solo contribuisce a migliorare le capacità di risposta agli incidenti, ma aiuta anche a prevenire il ripetersi dell'incidente. Imparando da ogni incidente, puoi evitare di ripetere gli errori, i rischi o le configurazioni non valide, non solo migliorando il tuo livello di sicurezza, ma anche riducendo al minimo il tempo speso in situazioni evitabili.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

È importante implementare un framework basato sulle lezioni apprese che stabilisce e raggiunge, ad alto livello, i seguenti punti:

- Quando si tiene un framework basato sulle lezioni apprese?
- Cosa comporta il processo basato sulle lezioni apprese?
- Come viene eseguito un framework basato sulle lezioni apprese?
- Chi è coinvolto nel processo e in che modo?
- Come vengono identificate le aree di miglioramento?
- In che modo garantisci che i miglioramenti vengano monitorati e implementati in modo efficace?

Il framework non deve concentrarsi sugli individui, ma sul miglioramento di strumenti e processi.

Passaggi dell'implementazione

A parte i risultati di alto livello sopra elencati, è importante porsi le domande giuste per trarre il massimo valore (informazioni che portano a miglioramenti attuabili) dal processo. Considera queste domande per iniziare a promuovere le discussioni sulle lezioni apprese:

- Qual è stato l'incidente?
- Quando è stato identificato per la prima volta l'incidente?
- Come è stato identificato?
- Quali sistemi hanno avvisato dell'attività?
- Quali sistemi, servizi e dati sono stati coinvolti?
- Cosa è successo nello specifico?
- Cosa ha funzionato bene?
- Cosa non ha funzionato bene?
- Quale processo o quali procedure non sono riusciti a dimensionare per rispondere all'incidente?
- Cosa può essere migliorato nelle seguenti aree:
  - Persone
    - Le persone da contattare erano effettivamente disponibili e l'elenco dei contatti era aggiornato?
    - Il personale presentava lacune nella formazione o nelle capacità necessarie per rispondere e indagare efficacemente sull'incidente?
    - Le risorse appropriate erano pronte e disponibili?
  - Elaborazione
    - Sono stati seguiti i processi e le procedure?
    - I processi e le procedure erano documentati e disponibili per questo tipo di incidente?
    - Mancavano i processi e le procedure richiesti?
    - Il team di risposta è stato in grado di accedere tempestivamente alle informazioni necessarie per rispondere al problema?
  - Tecnologia
    - I sistemi di avviso esistenti hanno identificato e segnalato efficacemente l'attività?
    - Come si sarebbe potuto ridurre il tempo di rilevamento del 50%?

- Gli avvisi esistenti devono essere migliorati o è necessario creare nuovi avvisi per questo tipo di incidente?
- Gli strumenti esistenti hanno consentito un'indagine efficace (ricerca/analisi) dell'incidente?
- Cosa si può fare per identificare prima questo tipo di incidente?
- Cosa si può fare per evitare che questo tipo di incidente si ripeta?
- A chi appartiene il piano di miglioramento e come verifichi che sia stato implementato?
- Qual è la tempistica per l'implementazione e il test del monitoraggio aggiuntivo o dei controlli e dei processi preventivi?

Questo elenco non è esaustivo, ma può fungere da punto di partenza per individuare quali sono le esigenze dell'organizzazione e dell'attività e come analizzarle per imparare in modo più efficace dagli incidenti e migliorare costantemente il proprio livello di sicurezza. La cosa più importante è iniziare incorporando le lezioni apprese come parte standard del processo di risposta agli incidenti, della documentazione e delle aspettative di tutti gli stakeholder.

Risorse

Documenti correlati:

- AWS Security Incident Response Guide - Establish a framework for learning from incidents
- NCSC CAF guidance - Lessons learned

Sicurezza delle applicazioni

Domanda

- SEC 11. Come si incorporano e convalidano le proprietà di sicurezza delle applicazioni nell'intero ciclo di vita di progettazione, sviluppo e implementazione?

SEC 11. Come si incorporano e convalidano le proprietà di sicurezza delle applicazioni nell'intero ciclo di vita di progettazione, sviluppo e implementazione?

La formazione del personale, l'esecuzione di test tramite automazione, l'identificazione delle dipendenze e la convalida delle proprietà di sicurezza di strumenti e applicazioni riducono la probabilità del verificarsi di problemi di sicurezza nei carichi di lavoro di produzione.

Best practice

- SEC11-BP01 Formazione per la sicurezza delle applicazioni
- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test
- SEC11-BP03 Esecuzione di test di penetrazione (pen-test) a intervalli regolari
- SEC11-BP04 Revisioni manuali del codice
- SEC11-BP05 Centralizzazione dei servizi per pacchetti e dipendenze
- SEC11-BP06 Implementazione programmatica del software
- SEC11-BP07 Valutazione regolare delle proprietà di sicurezza delle pipeline
- SEC11-BP08 Creazione di un programma per l'integrazione della titolarità della sicurezza nei team responsabili del carico di lavoro

SEC11-BP01 Formazione per la sicurezza delle applicazioni

Fornisci formazione sulle procedure comuni agli sviluppatori nell'organizzazione in modo da garantire la sicurezza dello sviluppo e del funzionamento delle applicazioni. L'adozione di procedure di sviluppo incentrate sulla sicurezza riduce la probabilità di riscontrare problemi solo nella fase di revisione della sicurezza.

Risultato desiderato: progettazione del software tenendo conto della sicurezza. Quando gli sviluppatori in un'organizzazione ricevono formazione su procedure di sviluppo sicure iniziando con un modello di rischio, la qualità e la sicurezza complessive del software prodotto sono migliori. Questo approccio può ridurre il tempo necessario per distribuire il software o le funzionalità, in quanto saranno necessarie meno correzioni dopo la fase di revisione della sicurezza.

Ai fini di questa best practice, il concetto di sviluppo sicuro si riferisce al software scritto e agli strumenti o ai sistemi che supportano il ciclo di vita di sviluppo del software.

Anti-pattern comuni:

- Valutazione delle proprietà di sicurezza di un sistema solo in fase di revisione della sicurezza.
- Assegnazione di tutte le decisioni in materia di sicurezza al team responsabile della sicurezza.
- Mancata comunicazione della correlazione tra le decisioni adottate durante il ciclo di vita di sviluppo del software e le aspettative o policy complessive dell'organizzazione.
- Svolgimento del processo di revisione della sicurezza in una fase troppo tardiva.

Vantaggi dell'adozione di questa best practice:

- Migliore identificazione dei requisiti aziendali per la sicurezza all'inizio del ciclo di sviluppo.
- Capacità di identificare e correggere più rapidamente possibili problemi di sicurezza, per una distribuzione più rapida delle funzionalità.
- Migliore qualità del software e dei sistemi.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Fornisci formazione agli sviluppatori nell'organizzazione. Un corso iniziale sulla modellazione delle minacce è un'ottima base per la formazione sulla sicurezza. Idealmente, gli sviluppatori devono poter accedere in modalità self-service a informazioni pertinenti ai propri carichi di lavoro. Questo accesso può aiutarli a prendere decisioni informate sulle proprietà di sicurezza dei sistemi sviluppati senza dover chiedere a un altro team. Il processo di coinvolgimento del team responsabile della sicurezza nelle revisioni deve essere definito chiaramente e facile da seguire. Le fasi del processo di revisione devono essere incluse nella formazione sulla sicurezza. Quando sono disponibili modelli o schemi di implementazione noti, devono essere facili da trovare e collegare ai requisiti complessivi per la sicurezza. Valuta se usare AWS CloudFormation, costrutti del AWS Cloud Development Kit (AWS CDK), il Service Catalog o altri strumenti di creazione di modelli per ridurre la necessità di configurazioni personalizzate.

Passaggi dell'implementazione

- Per iniziare, presenta agli sviluppatori un corso sulla modellazione delle minacce per creare ottime basi e abituarli a riflettere sulla sicurezza.
- Fornisci accesso a risorse di formazione AWS Training and Certification, per i diversi settori o per partner AWS.
- Fornisci formazione sul processo di revisione della sicurezza dell'organizzazione, che spieghi la suddivisione delle responsabilità tra il team responsabile della sicurezza, i team del carico di lavoro e altri stakeholder.
- Pubblica linee guida self-service su come soddisfare i requisiti di sicurezza, inclusi esempi e modelli di codice, se disponibili.
- Richiedi regolarmente ai team di sviluppo feedback sull'esperienza durante il processo di revisione della sicurezza e la formazione correlata e usalo per migliorare le procedure.
- Usa campagne di simulazione o bug bash per ridurre il numero di problemi e migliorare le competenze degli sviluppatori.

Risorse

Best practice correlate:

- SEC11-BP08 Creazione di un programma per l'integrazione della titolarità della sicurezza nei team responsabili del carico di lavoro

Documenti correlati:

- AWS Training and Certification
- Come riflettere sulla governance della sicurezza nel cloud
- Come accostarsi alla modellazione delle minacce
- Accelerazione della formazione – AWS Skills Guild

Video correlati:

- Sicurezza proattiva: considerazioni e approcci

Esempi correlati:

- Workshop sulla modellazione delle minacce
- Industry awareness for developers

Servizi correlati:

- AWS CloudFormation
- Costrutti del AWS Cloud Development Kit (AWS CDK) (AWS CDK)
- Service Catalog
- AWS BugBust

SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test

Automatizza i test per le proprietà di sicurezza lungo il ciclo di vita di sviluppo e test. L'automazione semplifica l'identificazione coerente e ripetibile dei potenziali problemi nel software prima del rilascio, riducendo il rischio di riscontrare problemi di sicurezza nel software fornito.

Risultato desiderato: l'obiettivo dei test automatici è fornire un metodo programmatico per rilevare inizialmente e regolarmente i potenziali problemi lungo l'intero ciclo di vita di sviluppo. Automatizzando i test di regressione, puoi ripetere l'esecuzione di test funzionali e non funzionali per verificare che il software testato in precedenza continui ad avere le prestazioni previste dopo una modifica. Quando definisci unit test di sicurezza per verificare la presenza di configurazioni errate comuni, come autorizzazioni non corrette o mancanti, puoi identificare e correggere i problemi all'inizio del processo di sviluppo.

Per l'automazione dei test vengono usati test case dedicati per la convalida delle applicazioni, in base ai requisiti e alle funzionalità desiderate. Il risultato dei test automatici è basato sul confronto dell'output di test generato con quello previsto, che accelera l'intero ciclo di vita dei test. Metodologie di test come i test di regressione e le suite di unit test sono le più adatte per l'automazione. L'automazione dei test delle proprietà di sicurezza permette agli sviluppatori di ricevere automaticamente feedback senza attendere una revisione della sicurezza. I test automatici sotto forma di analisi statica o dinamica del codice possono migliorare la qualità del codice e semplificare il rilevamento dei potenziali problemi software all'inizio del ciclo di vita di sviluppo.

Anti-pattern comuni:

- Mancata comunicazione dei test case e dei risultati dei test automatici.
- Esecuzione dei test solo immediatamente prima di un rilascio.
- Automazione dei test case con requisiti che cambiano spesso.
- Assenza di linee guida su come gestire i risultati dei test di sicurezza.

Vantaggi dell'adozione di questa best practice:

- Riduzione della dipendenza da valutazioni personali delle proprietà di sicurezza dei sistemi.
- Migliore coerenza grazie a risultati uniformi tra più flussi di lavoro.
- Minore probabilità di introdurre problemi di sicurezza nel software di produzione.
- Intervallo di tempo più breve tra il rilevamento e la correzione grazie all'identificazione più tempestiva dei problemi software.
- Maggiore visibilità su comportamenti sistematici o ripetuti tra più flussi di lavoro, che può essere usata per favorire miglioramenti in tutta l'organizzazione.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Durante lo sviluppo del software, adotta diversi meccanismi di test in modo da avere la certezza di testare l'applicazione per requisiti funzionali, basati sulla logica di business, e non funzionali, incentrati sull'affidabilità, sulle prestazioni e sulla sicurezza dell'applicazione.

I test di sicurezza statici dell'applicazione analizzano il codice sorgente in cerca di modelli di sicurezza anomali e forniscono indicazioni per un codice privo di errori. I test di sicurezza statici dell'applicazione si basano su input statici, come la documentazione (definizione dei requisiti, documentazione sulla progettazione e specifiche di progettazione) e il codice sorgente dell'applicazione, per testare un'ampia gamma di problemi di sicurezza noti. Gli analizzatori di codice statici possono contribuire ad accelerare l'analisi di volumi elevati di codice. Il NIST Quality Group offre un confronto tra gli analizzatori della sicurezza del codice sorgente, con strumenti open source per la scansione del codice byte e la scansione del codice binario.

Integra i test statici con metodologie di test della sicurezza tramite analisi dinamica, che eseguono test sull'applicazione in esecuzione per identificare potenziali comportamenti imprevisti. I test dinamici possono essere usati per individuare potenziali problemi non rilevabili tramite l'analisi statica. L'esecuzione di test nelle fasi di repository, compilazione e pipeline del codice permette di verificare potenziali problemi di tipi diversi, evitandone la presenza nel codice. Amazon CodeWhisperer fornisce suggerimenti per il codice, tra cui analisi della sicurezza, nell'ambiente di sviluppo integrato (IDE) dello sviluppatore. Il Amazon CodeGuru Reviewer può identificare problemi critici e di sicurezza e bug difficili da individuare durante lo sviluppo delle applicazioni e fornisce suggerimenti per migliorare la qualità del codice.

Il workshop sulla sicurezza per gli sviluppatori usa strumenti di sviluppo AWS come AWS CodeBuild, AWS CodeCommit eAWS CodePipeline per l'automazione della pipeline di rilascio, che include metodologie di test tramite analisi statiche e dinamiche.

Lungo il ciclo di vita di sviluppo del software definisci un processo iterativo che includa revisioni periodiche dell'applicazione con il team responsabile della sicurezza. Il feedback raccolto da queste revisioni della sicurezza deve essere affrontato e convalidato come parte della revisione dell'idoneità per il rilascio. Queste revisioni permettono di stabilire una solida posizione di sicurezza per l'applicazione e forniscono agli sviluppatori feedback di utilità pratica per affrontare i potenziali problemi.

Passaggi dell'implementazione

- Implementa un ambiente IDE, una revisione del codice e strumenti CI/CD coerenti che includano test di sicurezza.

- Determina le fasi del ciclo di vita di sviluppo del software in cui è appropriato bloccare le pipeline anziché informare semplicemente gli sviluppatori riguardo alla necessità di risolvere i problemi.
- Il workshop sulla sicurezza per gli sviluppatori fornisce un esempio di integrazione di test statici e dinamici in una pipeline di rilascio.
- L'esecuzione di test o di analisi del codice tramite strumenti automatici, come Amazon CodeWhisperer integrato con gli ambienti IDE degli sviluppatori e il Amazon CodeGuru Reviewer per l'analisi del codice in fase di commit, permette agli sviluppatori di ottenere feedback tempestivo.
- Se sviluppi soluzioni usando AWS Lambda, puoi usare Amazon Inspector per analizzare il codice dell'applicazione nelle funzioni.
- Il workshop sull'integrazione continua e sulla distribuzione continua in AWS fornisce un punto di partenza per la creazione di pipeline CI/CD in AWS.
- Quando le pipeline CI/CD includono test automatici, devi usare un sistema di gestione dei ticket per tenere traccia della notifica e della correzione dei problemi software.
- Per test di sicurezza che possono generare risultati, il collegamento a linee guida per la correzione permette agli sviluppatori di migliorare la qualità del codice.
- Analizza regolarmente i risultati ottenuti dagli strumenti automatici per definire le priorità delle successive iniziative di automazione, formazione degli sviluppatori o creazione di campagne di sensibilizzazione.

Risorse

Documenti correlati:

- Distribuzione e implementazione continue

- Partner con competenze in AWS DevOps
- Partner con competenze nella sicurezza AWS per la sicurezza delle applicazioni
- Scelta di un approccio CI/CD Well-Architected
- Monitoraggio di eventi CodeCommit in Amazon EventBridge e Amazon CloudWatch Events
- Rilevamento dei segreti nel revisore Amazon CodeGuru
- Accelerazione delle implementazioni su AWS con una governance efficace
- Approccio di AWS all'automazione di implementazioni pratiche e sicure

Video correlati:

- Informazioni pratiche: automazione di pipeline di distribuzione continua in Amazon
- Automazione di pipeline CI/CD tra account

Esempi correlati:

- Industry awareness for developers
- AWS CodePipeline Governance (GitHub)
- Workshop sulla sicurezza per gli sviluppatori
- Workshop sull'integrazione continua e sulla distribuzione continua in AWS

SEC11-BP03 Esecuzione di test di penetrazione (pen-test) a intervalli regolari

Esegui regolarmente test di penetrazione (pen-test) sul software. Questo meccanismo permette di identificare i potenziali problemi che non possono essere rilevati tramite test automatici o la revisione manuale del codice. Può anche aiutarti a determinare l'efficacia dei controlli di rilevamento. I test di penetrazione (pen-test) devono determinare se il software può funzionare in modi imprevisti, ad esempio esponendo dati che devono essere protetti o concedendo autorizzazioni più elevate del previsto.

Risultato desiderato: uso di test di penetrazione (pen-test) per rilevare, correggere e convalidare le proprietà di sicurezza dell'applicazione. È necessario eseguire test di penetrazione (pen-test) regolari e pianificati come parte del ciclo di vita di sviluppo del software. I risultati ottenuti dai test di penetrazione (pen-test) devono essere affrontati prima del rilascio del software. Devi analizzare i risultati dei test di penetrazione (pen-test) per identificare se vi siano problemi che possono essere identificati con l'automazione. Un processo di esecuzione di test di penetrazione (pen-test) regolare e ripetibile che includa un meccanismo di feedback attivo aiuta a stabilire linee guida per gli sviluppatori e migliora la qualità del software.

Anti-pattern comuni:

- Esecuzione di test di penetrazione (pen-test) solo per problemi di sicurezza noti o comuni.
- Esecuzione di test di penetrazione (pen-test) delle applicazioni senza gli strumenti e le librerie di terze parti dipendenti.
- Esecuzione di test di penetrazione (pen-test) solo per i problemi di sicurezza relativi ai pacchetti, senza valutare la logica di business implementata.

Vantaggi dell'adozione di questa best practice:

- Maggiore certezza riguardo alle proprietà di sicurezza del software prima del rilascio.
- Opportunità di identificare i modelli comportamentali preferiti delle applicazioni, per una migliore qualità del software.
- Presenza di un ciclo di feedback che identifica all'inizio del ciclo di sviluppo i punti in cui l'automazione o una formazione aggiuntiva possono migliorare le proprietà di sicurezza del software.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

I test di penetrazione (pen-test) sono un esercizio strutturato per l'esecuzione di test di sicurezza in cui vengono eseguiti scenari di violazione della sicurezza pianificati per rilevare, correggere e convalidare i controlli di sicurezza. I test di penetrazione (pen-test) iniziano dalla ricognizione, durante la quale vengono raccolti dati in base all'attuale progettazione dell'applicazione e alle sue dipendenze. Viene creato ed eseguito un elenco selezionato di scenari di test specifici per la sicurezza. Lo scopo principale di questi test è rivelare i problemi di sicurezza nell'applicazione che potrebbero essere sfruttati per ottenere accesso accidentale all'ambiente o accesso non autorizzato ai dati. Devi eseguire test di penetrazione (pen-test) quando avvii nuove funzionalità o ogni volta che l'applicazione viene sottoposta a modifiche importanti durante l'implementazione tecnica o di funzioni.

Devi identificare la fase più appropriata del ciclo di vita di sviluppo in cui eseguire i test di penetrazione (pen-test). Questi test devono essere eseguiti nelle fasi finali, in modo che la funzionalità del sistema sia vicina allo stato di rilascio previsto, ma con tempo sufficiente per la correzione di eventuali problemi.

Passaggi dell'implementazione

- Prepara un processo strutturato per definire l'ambito dei test di penetrazione (pen-test). Un ottimo metodo per mantenere il contesto consiste nel basare questo processo sul modello di rischio.
- Identifica la fase più appropriata del ciclo di vita di sviluppo in cui eseguire test di penetrazione (pen-test). Questi devono avvenire quando sono previste modifiche minime nell'applicazione, ma quando vi è ancora tempo sufficiente per apportare eventuali correzioni.
- Prepara gli sviluppatori su cosa aspettarsi dai risultati dei test di penetrazione (pen-test) e su come ottenere informazioni sulla correzione.

- Usa strumenti per accelerare il processo di esecuzione dei test di penetrazione (pen-test) automatizzando test comuni o ripetibili.
- Analizza i risultati dei test di penetrazione (pen-test) per identificare problemi di sicurezza sistematici e usa questi dati per definire altri test automatici e formazione continua per gli sviluppatori.

Risorse

Best practice correlate:

- SEC11-BP01 Formazione per la sicurezza delle applicazioni
- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test

Documenti correlati:

- La pagina Test di penetrazione (pen-test) AWS fornisce linee guida dettagliate per l'esecuzione di test di penetrazione (pen-test) in AWS
- Accelerazione delle implementazioni su AWS con una governance efficace
- Partner con competenze nella sicurezza AWS
- Modernizzazione dell'architettura dei test di penetrazione (pen-test) su AWS Fargate
- Simulatore di iniezione guasti AWS

Esempi correlati:

- Automate API testing with AWS CodePipeline (GitHub)
- Automated security helper (GitHub)

SEC11-BP04 Revisioni manuali del codice

Esegui una revisione manuale del codice del software che produci. Attraverso questo processo puoi assicurarti che chi ha scritto il codice non sia l'unica persona a controllarne la qualità.

Risultato desiderato: l'aggiunta di una fase di revisione manuale del codice durante lo sviluppo migliora la qualità del software scritto, permette di affinare le competenze dei membri meno esperti del team e fornisce un'opportunità per identificare i punti in cui può essere usata l'automazione. Le revisioni manuali del codice possono essere supportate da strumenti e test automatici.

Anti-pattern comuni:

- Non viene eseguita alcuna revisione del codice prima dell'implementazione.
- La scrittura e la revisione del codice vengono effettuate dalla stessa persona.
- Non viene usata l'automazione per semplificare o orchestrare le revisioni del codice.
- Gli sviluppatori non ricevono formazione sulla sicurezza dell'applicazione prima di eseguire la revisione del codice.

Vantaggi dell'adozione di questa best practice:

- Migliore qualità del codice.
- Maggiore coerenza dello sviluppo del codice attraverso il riutilizzo di approcci comuni.
- Riduzione del numero di problemi riscontrati durante i test di penetrazione e nelle fasi successive.
- Migliore circolazione delle informazioni all'interno del team.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

La fase di revisione deve essere implementata come parte del flusso complessivo di gestione del codice. Le specifiche dipendono dall'approccio usato per la diramazione, le richieste pull e l'unione. Puoi usare AWS CodeCommit o soluzioni di terze parti come GitHub, GitLab o Bitbucket. Qualunque sia il metodo usato, è importante verificare che i processi richiedano la revisione del codice prima che venga implementato in un ambiente di produzione. L'uso di strumenti come il Amazon CodeGuru Reviewer può semplificare l'orchestrazione del processo di revisione del codice.

Passaggi dell'implementazione

- Implementa una fase di revisione manuale come parte del flusso di gestione del codice ed esegui la revisione prima di continuare.
- Prendi in considerazione il Amazon CodeGuru Reviewer per la gestione e la semplificazione delle revisioni del codice.
- Implementa un flusso di approvazione che richieda il completamento di una revisione prima che il codice possa passare alla fase successiva.
- Verifica che sia stato definito un processo per identificare i problemi riscontrati durante le revisioni manuali del codice che potrebbero essere rilevati automaticamente.

- Integra la fase di revisione manuale del codice in modo che sia allineata alla procedure di sviluppo del codice.

Risorse

Best practice correlate:

- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test

Documenti correlati:

- Uso di richieste pull in repository AWS CodeCommit
- Uso di modelli per le regole di approvazione in AWS CodeCommit
- GitHub: About pull requests
- Automazione delle revisioni del codice con il Amazon CodeGuru Reviewer
- Automazione del rilevamento di bug e vulnerabilità della sicurezza in pipeline CI/CD usando l'interfaccia della riga di comando del Amazon CodeGuru Reviewer

Video correlati:

- Miglioramento continuo della qualità del codice con Amazon CodeGuru

Esempi correlati:

- Workshop sulla sicurezza per gli sviluppatori

SEC11-BP05 Centralizzazione dei servizi per pacchetti e dipendenze

Fornisci servizi centralizzati per permettere ai team di sviluppo di ottenere pacchetti software e altre dipendenze. Questo approccio permette la convalida dei pacchetti prima di includerli nel software scritto e fornisce un'origine dati per l'analisi del software usato nell'organizzazione.

Risultato desiderato: il software include una serie di altri pacchetti software oltre al codice scritto. In questo modo, è più facile utilizzare implementazioni di funzionalità usate ripetutamente, come un parser JSON o una libreria di crittografia. La centralizzazione logica delle origini per questi pacchetti e dipendenze fornisce un meccanismo tramite il quale i team responsabili della sicurezza possono

convalidare le proprietà dei pacchetti prima che vengano usati. Questo approccio riduce anche il rischio di un problema imprevisto causato da una modifica in un pacchetto esistente o dall'aggiunta da parte dei team di sviluppo di pacchetti arbitrari direttamente da Internet. Usa questo approccio insieme ai flussi di test manuali e automatici per garantire ulteriormente la qualità del software sviluppato.

Anti-pattern comuni:

- Recupero di pacchetti da repository arbitrari su Internet.
- Mancata esecuzione di test sui nuovi pacchetti prima di renderli disponibili agli sviluppatori.

Vantaggi dell'adozione di questa best practice:

- Migliore comprensione dei pacchetti usati nel software sviluppato.
- Capacità di informare i team responsabili del carico di lavoro quando un pacchetto deve essere aggiornato in base alle informazioni su chi usa cosa.
- Minor rischio di includere nel software un pacchetto con problemi.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Fornisci servizi centralizzati per i pacchetti e le dipendenze in modo da semplificarne l'uso per gli sviluppatori. La centralizzazione dei servizi può essere eseguita in modo logico anziché implementarli come sistema monolitico. Questo approccio permette di fornire servizi in modo da soddisfare le esigenze degli sviluppatori. Ti consigliamo di implementare un metodo efficiente per aggiungere pacchetti al repository quando sono necessari aggiornamenti o emergono nuovi requisiti. Servizi AWS come AWS CodeArtifact o soluzioni simili di partner AWS forniscono alcuni strumenti utili per questo scopo.

Passaggi dell'implementazione:

- Implementa un servizio di repository centralizzato in modo logico che sia disponibile in tutti gli ambienti in cui viene sviluppato il software.
- Includi l'accesso al repository come parte del processo di provisioning automatico dell'Account AWS.
- Crea automazione per testare i pacchetti prima che vengano pubblicati in un repository.

- Gestisci le metriche dei pacchetti, dei linguaggi e dei team usati più comunemente e con la maggiore quantità di modifiche.
- Offri ai team di sviluppo un meccanismo automatico per richiedere nuovi pacchetti e fornire feedback.
- Analizza regolarmente i pacchetti nel repository per identificare il possibile impatto di nuovi problemi riscontrati.

Risorse

Best practice correlate:

- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test

Documenti correlati:

- Accelerazione delle implementazioni su AWS con una governance efficace
- Potenziamento della sicurezza dei pacchetti con il toolkit per il controllo delle origini dei pacchetti CodeArtifact
- Rilevamento dei problemi di sicurezza nella registrazione con il Amazon CodeGuru Reviewer
- Supply chain Levels for Software Artifacts (SLSA)

Video correlati:

- Sicurezza proattiva: considerazioni e approcci
- La filosofia di AWS per la sicurezza (re:Invent 2017)
- Quando sicurezza, protezione e urgenza sono tutte importanti: gestione di Log4Shell

Esempi correlati:

- Multi Region Package Publishing Pipeline (GitHub)
- Publishing Node.js Modules on AWS CodeArtifact using AWS CodePipeline (GitHub)
- AWS CDK Java CodeArtifact Pipeline Sample (GitHub)
- Distribute private .NET NuGet packages with AWS CodeArtifact (GitHub)

SEC11-BP06 Implementazione programmatica del software

Esegui implementazioni programmatiche del software laddove possibile. Questo approccio riduce la probabilità che un'implementazione non riesca o che si verifichi un problema imprevisto a causa dell'errore umano.

Risultato desiderato: un intervento minimo sui dati da parte delle persone è un principio chiave dello sviluppo sicuro nel Cloud AWS. Questo principio include anche il modo in cui viene implementato il software.

I vantaggi legati alla scelta di non affidare a persone l'implementazione del software è la migliore garanzia che la soluzione implementata sia esattamente identica a quella testata e che l'implementazione verrà eseguita in modo coerente ogni volta. Il software non deve essere modificato in modo da funzionare in ambienti diversi. Usando i principi dello sviluppo di applicazioni a dodici fattori, in particolare l'esternalizzazione della configurazione, puoi implementare lo stesso codice in più ambienti senza richiedere modifiche. La firma crittografica dei pacchetti software è un ottimo metodo per verificare che non vengano apportate modifiche tra ambienti. Il risultato complessivo di questo approccio è la riduzione dei rischi nel processo di modifica e il miglioramento della coerenza delle versioni del software.

Anti-pattern comuni:

- Implementazione manuale del software nell'ambiente di produzione.
- Applicazione manuale di modifiche al software per soddisfare i requisiti di ambienti diversi.

Vantaggi dell'adozione di questa best practice:

- Maggiore affidabilità del processo di rilascio del software.
- Riduzione dei rischi legati a modifiche errate che hanno impatto sulla funzionalità aziendale.
- Processi di rilascio più frequenti grazie a un rischio di modifica minimo.
- Funzionalità di ripristino automatico dello stato precedente in caso di eventi imprevisti durante l'implementazione.
- Possibilità di usare la crittografia per dimostrare che il software implementato è esattamente identico a quello testato.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

Crea la struttura di Account AWS per eliminare l'accesso umano frequente dagli ambienti e usa strumenti CI/CD per eseguire le implementazioni. Progetta le applicazioni in modo da ottenere i dati di configurazione specifici dell'ambiente da un'origine esterna, ad esempio l'Archivio dei parametri AWS Systems Manager. Firma i pacchetti dopo che vengono testati e convalida le firme durante l'implementazione. Configura le pipeline CI/CD per il push del codice delle applicazioni e usa valori Canary per confermare la corretta esecuzione dell'implementazione. Usa strumenti come AWS CloudFormation o il AWS CDK per definire l'infrastruttura, quindi AWS CodeBuild e AWS CodePipeline per eseguire operazioni CI/CD.

Passaggi dell'implementazione

- Crea pipeline CI/CD ben definite per semplificare il processo di implementazione.
- L'uso di AWS CodeBuild e AWS Code Pipeline per fornire funzionalità CI/CD semplifica l'integrazione di test di sicurezza nelle pipeline.
- Segui le linee guida sulla separazione degli ambienti nel whitepaper sull'organizzazione dell'ambiente AWS usando più account.
- Verifica che non si verifichi accesso umano frequente agli ambienti in cui sono in esecuzione carichi di lavoro di produzione.
- Progetta le applicazioni in modo che supportino l'esternalizzazione dei dati di configurazione.
- Valuta se eseguire l'implementazione usando un modello di implementazione blu/verde.
- Implementa valori Canary per convalidare la corretta implementazione del software.
- Usa strumenti di crittografia come AWS Signer o AWS Key Management Service (AWS KMS) per firmare e verificare i pacchetti software implementati.

Risorse

Best practice correlate:

- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test

Documenti correlati:

- Workshop sull'integrazione continua e sulla distribuzione continua in AWS
- Accelerazione delle implementazioni su AWS con una governance efficace

- Automazione di implementazioni pratiche e sicure
- Firma del codice usando l'Autorità privata per la gestione del certificato AWS (ACM CA privata/ACM PCA) e chiavi asimmetriche AWS Key Management Service
- Firma del codice, un controllo di attendibilità e integrità per AWS Lambda

Video correlati:

- Informazioni pratiche: automazione di pipeline di distribuzione continua in Amazon

Esempi correlati:

- Implementazioni blu/verde con AWS Fargate

SEC11-BP07 Valutazione regolare delle proprietà di sicurezza delle pipeline

Applica i principi della sicurezza Well-Architected alle pipeline, con particolare attenzione alla separazione delle autorizzazioni. Valuta regolarmente le proprietà di sicurezza dell'infrastruttura di pipeline. Una gestione efficace della sicurezza delle pipeline assicura la protezione del software che passa attraverso le pipeline.

Risultato desiderato: le pipeline usate per sviluppare e implementare il software devono seguire le stesse procedure consigliate di qualsiasi altro carico di lavoro nell'ambiente. I test implementati nelle pipeline non devono essere modificabili dagli sviluppatori che li usano. Le pipeline devono avere solo le autorizzazioni necessarie per le implementazioni eseguite e devono applicare misure di protezione per evitare l'implementazione negli ambienti errati. Le pipeline non devono basarsi su credenziali a lungo termine e devono essere configurate in modo da emettere lo stato, per permettere la convalida dell'integrità degli ambienti di sviluppo.

Anti-pattern comuni:

- Test di sicurezza che possono essere ignorati dagli sviluppatori.
- Autorizzazioni eccessivamente elevate per le pipeline di implementazione.
- Pipeline non configurate per la convalida degli input.
- Nessuna revisione periodica delle autorizzazioni associate all'infrastruttura CI/CD.
- Uso di credenziali a lungo termine o hardcoded.

Vantaggi dell'adozione di questa best practice:

- Maggiore garanzia di integrità del software sviluppato e implementato attraverso le pipeline.
- Possibilità di arrestare un'implementazione in caso di attività sospetta.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

Iniziando con servizi CI/CD gestiti che supportano ruoli IAM, puoi ridurre il rischio di perdita di credenziali. L'applicazione dei principi della sicurezza all'infrastruttura di pipeline CI/CD può aiutarti a determinare i punti in cui apportare miglioramenti per la sicurezza. Un ottimo punto di partenza per la creazione degli ambienti CI/CD è l'architettura di riferimento per le pipeline di implementazione AWS. La revisione periodica dell'implementazione delle pipeline e l'analisi dei log per identificare comportamenti imprevisti può semplificare la comprensione dei modelli di utilizzo delle pipeline usate per implementare il software.

Passaggi dell'implementazione

- Inizia dall'architettura di riferimento per le pipeline di implementazione AWS.
- Valuta se usare il AWS IAM Access Analyzer per generare in modo programmatico policy IAM con privilegi minimi per le pipeline.
- Integra le pipeline con monitoraggio e generazione di avvisi in modo da ricevere notifiche in caso di attività imprevista o anomala, in quanto Amazon EventBridge per servizi gestiti AWS permette di instradare dati a destinazioni come AWS Lambda o Amazon Simple Notification Service (Amazon SNS).

Risorse

Documenti correlati:

- Architettura di riferimento per pipeline di implementazione AWS
- Monitoraggio di AWS CodePipeline
- Best practice per la sicurezza per AWS CodePipeline

Esempi correlati:

- DevOps monitoring dashboard (GitHub)

SEC11-BP08 Creazione di un programma per l'integrazione della titolarità della sicurezza nei team responsabili del carico di lavoro

Crea un programma o un meccanismo che permetta ai team di sviluppo di prendere decisioni sulla sicurezza del software che creano. Il team responsabile della sicurezza dovrà convalidare queste decisioni durante una revisione, ma l'integrazione della titolarità della sicurezza nei team di sviluppo permette di creare carichi di lavoro più veloci e sicuri. Questo meccanismo promuove anche una cultura della responsabilità che ha un impatto positivo sul funzionamento dei sistemi che crei.

Risultato desiderato: per integrare la titolarità della sicurezza e il processo decisionale nei team di sviluppo, puoi insegnare agli sviluppatori come riflettere sulla sicurezza o puoi migliorarne la formazione attraverso l'integrazione o l'associazione di responsabili della sicurezza nei team di sviluppo. Entrambi gli approcci sono validi e permettono al team di prendere decisioni di qualità migliore sulla sicurezza nelle fasi iniziali del ciclo di sviluppo. Questo modello di titolarità è basato sulla formazione per la sicurezza delle applicazioni. Iniziando dal modello di rischio per il carico di lavoro specifico, puoi concentrarti sul design thinking nel contesto appropriato. Un altro vantaggio della presenza di una comunità di sviluppatori attenti alla sicurezza o di un gruppo di tecnici della sicurezza che collabora con i team di sviluppo è la possibilità di comprendere a pieno il modo in cui è compilato il codice. Questa comprensione permette di determinare le aree di miglioramento successive per l'automazione.

Anti-pattern comuni:

- Assegnazione di tutte le decisioni in materia di sicurezza al team responsabile della sicurezza.
- Gestione dei requisiti di sicurezza in fasi tardive del processo di sviluppo.
- Assenza di feedback di sviluppatori e responsabili della sicurezza sul funzionamento del programma.

Vantaggi dell'adozione di questa best practice:

- Riduzione del tempo necessario per completare le revisioni della sicurezza.
- Riduzione dei problemi di sicurezza rilevati solo in fase di revisione della sicurezza.
- Miglioramento della qualità complessiva del software scritto.
- Opportunità di identificare e comprendere i problemi sistematici o le aree di miglioramento a valore elevato.
- Riduzione della quantità di attività di correzione dovute ai risultati delle revisioni della sicurezza.

- Migliore percezione della funzione della sicurezza.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

Per iniziare, segui le linee guida fornite in SEC11-BP01 Formazione per la sicurezza delle applicazioni. Identifica quindi il modello operativo per il programma che ritieni più efficace per l'organizzazione. I due modelli principali consistono nel formare gli sviluppatori o nell'integrare responsabili della sicurezza nei team di sviluppo. Una volta scelto l'approccio iniziale, devi eseguire un progetto pilota con un singolo team o un piccolo gruppo di team del carico di lavoro per dimostrare il funzionamento del modello per l'organizzazione. Il supporto autorevole da parte dello sviluppatore e di altre parti responsabili della sicurezza dell'organizzazione semplifica l'implementazione e il successo del programma. Durante la creazione del programma è importante scegliere metriche da usare per dimostrarne il valore. Per un'ottima esperienza formativa, puoi documentarti sul modo in cui AWS ha affrontato questo problema. Questa best practice è per lo più incentrata sulla trasformazione e sulla cultura aziendali. Gli strumenti usati devono supportare la collaborazione tra lo sviluppatore e le comunità responsabili della sicurezza.

Passaggi dell'implementazione

- Per iniziare, fornisci formazione sulla sicurezza delle applicazioni agli sviluppatori.
- Crea una comunità e un programma di onboarding per preparare gli sviluppatori.
- Scegli un nome per il programma. Alcuni termini comunemente usati sono Responsabilità, Supporto o Promozione.
- Identifica il modello da usare: formazione per gli sviluppatori, integrazione di tecnici della sicurezza o ruoli di sicurezza per affinità.
- Identifica alcuni sponsor del progetto tra responsabili della sicurezza, sviluppatori e altri gruppi potenzialmente rilevanti.
- Tieni traccia delle metriche per il numero di persone coinvolte nel programma, il tempo impiegato per le revisioni e il feedback ottenuto da sviluppatori e responsabili della sicurezza. Usa queste metriche per apportare miglioramenti.

Risorse

Best practice correlate:

- SEC11-BP01 Formazione per la sicurezza delle applicazioni
- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test

Documenti correlati:

- Come accostarsi alla modellazione delle minacce
- Come riflettere sulla governance della sicurezza nel cloud

Video correlati:

- Sicurezza proattiva: considerazioni e approcci

Affidabilità

Il principio dell'affidabilità comprende la capacità di un carico di lavoro di eseguire la funzione attesa in modo corretto e coerente quando previsto. È possibile trovare linee guida prescrittive sull'implementazione nel Whitepaper sul principio dell'affidabilità.

Aree delle best practice

- Fondamenti
- Architettura del carico di lavoro
- Gestione delle modifiche
- Gestione degli errori

Fondamenti

Domande

- REL 1. Come si gestiscono quote e vincoli di servizio?
- REL 2. Come si pianifica la topologia di rete?

REL 1. Come si gestiscono quote e vincoli di servizio?

Per le architetture di carichi di lavoro basate sul cloud, esistono quote di servizio (definite anche come restrizioni dei servizi). Queste quote sono presenti per evitare di effettuare accidentalmente

il provisioning di più risorse di quelle necessarie e limitare i tassi di richiesta sulle operazioni API in modo da proteggere i servizi da un uso illecito. Esistono anche vincoli di risorse, ad esempio la velocità con cui è possibile trasferire i bit su un cavo in fibra ottica o la quantità di storage su un disco fisico.

Best practice

- REL01-BP01 Consapevolezza su quote e vincoli di servizio
- REL01-BP02 Gestione delle quote di servizio in più account e regioni
- REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura
- REL01-BP04 Monitoraggio e gestione delle quote
- REL01-BP05 Automazione della gestione delle quote
- REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover

REL01-BP01 Consapevolezza su quote e vincoli di servizio

Conosci le quote predefinite e gestisci le richieste di aumento delle quote per l'architettura del carico di lavoro. Sai quali vincoli delle risorse cloud, ad esempio disco o rete, sono potenzialmente influenti.

Risultato desiderato: i clienti possono evitare il degrado dei servizi o l'interruzione in Account AWS implementando linee guida appropriate per il monitoraggio di metriche chiave, revisioni dell'infrastruttura e fasi di correzione dell'automazione per verificare che non vengano raggiunte quote e vincoli dei servizi che potrebbero causare degrado o interruzione del servizio.

Anti-pattern comuni:

- Distribuzione di un carico di lavoro senza comprendere le quote hard o soft e i relativi limiti per i servizi utilizzati.
- Distribuzione di un carico di lavoro sostitutivo senza analizzare e riconfigurare le quote necessarie o contattare preventivamente l'assistenza.
- Supposizione che i servizi cloud non abbiano limiti e che i servizi possano essere utilizzati senza tener conto di tariffe, limiti, conteggi, quantità.
- Supposizione che le quote verranno aumentate automaticamente.
- Mancata conoscenza del processo e della scadenza delle richieste di quote.
- Supposizione che la quota predefinita del servizio cloud sia identica per ogni servizio rispetto alle varie regioni.

- Supposizione che i vincoli del servizio possano essere violati e che i sistemi si autoscalino o aumentino il limite oltre i vincoli della risorsa
- Nessun test dell'applicazione nei momenti di picco del traffico, per stressare l'utilizzo delle sue risorse.
- Provisioning della risorsa senza analisi della dimensione della risorsa richiesta.
- Provisioning in eccesso di capacità scegliendo tipi di risorse che vanno ben oltre il fabbisogno effettivo o i picchi previsti.
- Nessuna valutazione dei requisiti di capacità per nuovi livelli di traffico prima di un nuovo evento cliente o dell'implementazione di una nuova tecnologia.

Vantaggi derivanti dall'adozione di questa best practice: monitoraggio e gestione automatizzata delle quote di servizio e limiti delle risorse possono ridurre i guasti in modo proattivo. I cambiamenti nei modelli di traffico per il servizio di un cliente possono causare un'interruzione o un degrado se non si seguono le best practice. Monitorando e gestendo questi valori in tutte le regioni e in tutti gli account, le applicazioni possono avere una maggiore resilienza in caso di eventi avversi o non pianificati.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

Service Quotas è un servizio AWS che ti aiuta a gestire le quote per oltre 250 servizi AWS da un'unica posizione. Oltre a cercare i valori delle quote, si possono anche richiedere e monitorare gli aumenti delle quote stesse tramite la console Service Quotas o tramite l'SDK AWS. AWS Trusted Advisor offre un controllo delle quote di servizio che mostra l'utilizzo e le quote per alcuni aspetti di determinati servizi. Le quote di servizio predefinite per servizio sono indicate anche nella documentazione AWS di ciascun servizio (ad esempio vedi Quote di Amazon VPC).

Alcuni limiti dei servizi, come i limiti di velocità sulle API con throttling vengono impostati all'interno di Amazon API Gateway stesso configurando un piano di utilizzo. Altri limiti impostati come configurazione per i rispettivi servizi includono capacità di IOPS allocata, archiviazione Amazon RDS allocato e allocazioni di volumi Amazon EBS. Amazon Elastic Compute Cloud dispone di un proprio pannello di controllo sui limiti del servizio che consente di gestire l'istanza, Amazon Elastic Block Store e i limiti degli indirizzi IP elastici. Se hai un caso d'uso in cui le quote di servizio influiscono sulle prestazioni della tua applicazione e non sono adattabili alle tue esigenze, contatta AWS Support per vedere se sono possibili riduzioni.

Le quote di servizio possono essere specifiche per ogni regione o di natura globale. L'uso di un servizio AWS che raggiunge la sua quota non si comporterà come previsto nell'uso normale e

potrebbe causare interruzioni o degrado del servizio. Ad esempio, una quota di servizio limita il numero di DL Amazon EC2 che può essere usato in una Regione e tale limite può essere raggiunto durante un evento di dimensionamento del traffico tramite gruppi Auto Scaling (ASG).

Le quote di servizio per ogni account devono essere valutate regolarmente per determinare quali siano i limiti di servizio appropriati per quell'account. Queste quote di servizio esistono come guardrail operativi, per evitare di fornire accidentalmente più risorse di quelle necessarie. Servono anche a limitare i tassi di richiesta delle operazioni API per proteggere i servizi dagli abusi.

I limiti dei servizi sono diversi dalle quote dei servizi. I vincoli di servizio rappresentano i limiti di una particolare risorsa, definiti da quel tipo di risorsa. Questi possono essere la capacità di archiviazione (ad esempio, gp2 ha un limite di dimensione di 1 GB - 16 TB) o il throughput del disco (10.0000 iops). È essenziale che il vincolo di un tipo di risorsa sia progettato e valutato costantemente per l'utilizzo che potrebbe raggiungere il suo limite. Se un vincolo viene raggiunto inaspettatamente, le applicazioni o i servizi dell'account possono essere degradati o interrotti.

Se hai un caso d'uso in cui le quote di servizio influiscono sulle prestazioni della tua applicazione e non sono adattabili alle tue esigenze, contatta AWS Support per vedere se sono possibili mitigazioni. Per maggiori dettagli su come modificare le quote fisse vedi REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura.

Esistono alcuni servizi e strumenti AWS per monitorare e gestire Service Quotas. Il servizio e gli strumenti devono essere sfruttati per fornire controlli automatici o manuali dei livelli di quota.

- AWS Trusted Advisor offre un controllo delle quote di servizio che mostra l'utilizzo e le quote per alcuni aspetti di alcuni servizi. Può aiutare a identificare i servizi vicini alle quote.
- AWS Management Console fornisce metodi per visualizzare i valori delle quote dei servizi, gestire, richiedere nuove quote, monitorare lo stato delle richieste di quote e visualizzare la cronologia delle quote.
- AWS CLI e CDK offrono metodi programmatici per gestire e monitorare automaticamente l'utilizzo e i livelli delle quote di servizio.

Passaggi dell'implementazione

Per Service Quotas:

- Revisione di AWS Service Quotas.
- Per essere certo delle quote di servizio esistenti, stabilisci i servizi (come IAM Access Analyzer) usati. Esistono circa 250 servizi AWS controllati da quote di servizio. Quindi stabilisci il nome della

quota di servizio specifica che potrebbe essere usata all'interno di ogni account e regione. Esistono circa 3000 nomi di quote di servizio per regione.

- Aumenta questa analisi delle quote con AWS Config per trovare tutte le risorse AWS usate nei tuoi Account AWS.
- Utilizza i dati AWS CloudFormation per stabilire le risorse AWS utilizzate. Esamina le risorse create in AWS Management Console o con il comando `list-stack-resources` AWS CLI. È anche possibile vedere le risorse configurate da distribuire nel modello stesso.
- Stabilisci tutti i servizi necessari per il tuo carico di lavoro analizzando il codice di implementazione.
- Determina le quote di servizio applicabili. Utilizza le informazioni accessibili in modo programmatico da Trusted Advisor e Service Quotas.
- Stabilisci un metodo di monitoraggio automatizzato (vedi REL01-BP02 Gestione delle quote di servizio in più account e regioni e REL01-BP04 Monitoraggio e gestione delle quote) per avvisare e informare se le quote di servizio sono vicine o hanno raggiunto il limite.
- Stabilisci un metodo automatico e programmatico per verificare se una quota di servizio è stata modificata in una regione ma non in altre regioni dello stesso account. (consulta REL01-BP02 Gestione delle quote di servizio in più account e regioni e REL01-BP04 Monitoraggio e gestione delle quote).
- Automatizza la scansione dei log e delle metriche delle applicazioni per determinare se ci sono errori di quota o di vincoli di servizio. Se sono presenti errori, invia gli allarmi al sistema di monitoraggio.
- Stabilisci procedure di progettazione per calcolare la modifica richiesta nella quota (vedi REL01-BP05 Automazione della gestione delle quote) una volta stabilito che per alcuni servizi specifici sono richieste quote maggiori.
- Crea un flusso di lavoro di provisioning e di approvazione per richiedere modifiche alla quota di servizio. Questo dovrebbe includere un flusso di lavoro di eccezione in caso di rifiuto della richiesta o di approvazione parziale.
- Crea un metodo ingegneristico per rivedere le quote dei servizi prima del provisioning e dell'utilizzo di nuovi servizi AWS prima del roll-out in ambienti di produzione o carichi (ad esempio, account di test di carico).

Per i vincoli dei servizi:

- Stabilisci metodi di monitoraggio e metrica per avvisare se le risorse si avvicinano ai loro limiti. Sfrutta CloudWatch in base alle necessità per le metriche o il monitoraggio dei log.

- Stabilisci soglie di allarme per ogni risorsa che ha un vincolo significativo per l'applicazione o il sistema.
- Crea procedure di gestione del flusso di lavoro e dell'infrastruttura per cambiare il tipo di risorsa se il vincolo è prossimo all'utilizzo. Questo flusso di lavoro dovrebbe includere test di carico come best practice per verificare che quello nuovo sia il tipo di risorsa corretto in base ai nuovi vincoli.
- Migra la risorsa identificata al nuovo tipo di risorsa consigliato, utilizzando le procedure e i processi esistenti.

Risorse

Best practice correlate:

- REL01-BP02 Gestione delle quote di servizio in più account e regioni
- REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura
- REL01-BP04 Monitoraggio e gestione delle quote
- REL01-BP05 Automazione della gestione delle quote
- REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover
- REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL12-BP05 Test della resilienza tramite l'utilizzo dell'ingegneria del caos

Documenti correlati:

- Principio dell'affidabilità di AWS Well-Architected Framework: disponibilità
- AWS Service Quotas (precedentemente note come restrizioni dei servizi)
- Controlli delle best practice AWS Trusted Advisor (consulta la sezione Restrizioni dei servizi)
- Monitoraggio limite su AWS su risposte AWS
- Quote di servizio Amazon EC2
- Che cos'è Service Quotas?
- Come richiedere un aumento delle quote

- Endpoint e quote dei servizi
- Guida per l'utente di Service Quotas
- Monitoraggio delle quote per AWS
- Limiti di isolamento dei guasti di AWS
- Disponibilità con ridondanza
- AWS for Data
- In cosa consiste l'Integrazione continua?
- In cosa consiste la Distribuzione continua?
- Partner APN: partner per la gestione della configurazione
- Gestione del ciclo di vita dell'account in ambienti SaaS account-per-tenant su AWS
- Gestione e monitoraggio della limitazione delle API nei carichi di lavoro
- Visualizza i suggerimenti di AWS Trusted Advisor su scala con AWS Organizations
- Automazione dell'aumento dei limiti di servizio e supporto aziendale con AWS Control Tower

Video correlati:

- AWS Live re:Inforce 2019 - Service Quotas
- Visualizza e gestisci quote per i servizi AWS con Service Quotas
- Demo delle quote AWS IAM

Strumenti correlati:

- Amazon CodeGuru Reviewer
- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK

- AWS Systems Manager
- Marketplace AWS

REL01-BP02 Gestione delle quote di servizio in più account e regioni

Se utilizzi più account o Regioni, assicurati di richiedere le quote appropriate in tutti gli ambienti in cui vengono eseguiti i carichi di lavoro di produzione.

Risultato desiderato: i servizi e le applicazioni non dovrebbero essere interessati dall'esaurimento delle quote di servizio per le configurazioni che si estendono su account o Regioni o con progetti di resilienza che utilizzano il failover di zona, Regione o account.

Anti-pattern comuni:

- Consentire l'aumento dell'utilizzo delle risorse in una Regione di isolamento senza alcun meccanismo per mantenere la capacità nelle altre.
- Impostare manualmente tutte le quote in modo indipendente nelle Regioni di isolamento.
- Non considerare l'effetto delle architetture di resilienza (come quelle attive o passive) nelle future esigenze di quote durante un degrado nella Regione non primaria.
- Non valutare regolarmente le quote e apportare le modifiche necessarie in ogni Regione e account in cui viene gestito il carico di lavoro.
- Non sfruttare modelli di richiesta delle quote per richiedere incrementi su più Regioni e account.
- Non aggiornare le quote dei servizi, perché si pensa erroneamente che l'aumento delle quote abbia implicazioni di costo, come le richieste di prenotazione di calcolo.

Vantaggi derivanti dall'adozione di questa best practice: verificare che sia possibile gestire il proprio carico attuale in Regioni o account secondari in caso di mancata disponibilità dei servizi regionali. Questo consente di ridurre il numero di errori o livelli di degrado che si verificano durante la perdita di Regioni.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

Le quote di servizio vengono monitorate per account. Salvo diversa indicazione, ogni quota è specifica della Regione AWS. Oltre agli ambienti di produzione, gestisci anche le quote in tutti

gli ambienti non di produzione applicabili, in modo che i test e lo sviluppo non siano ostacolati. Il mantenimento di un elevato grado di resilienza richiede una valutazione continua delle quote di servizio (sia automatica che manuale).

Con più carichi di lavoro in diverse Regioni a causa dell'implementazione di progetti che usano approcci Active/Active, Active/Passive – Hot, Active/Passive-Cold e Active/Passive-Pilot Light, è fondamentale comprendere tutti i livelli di quote di account e Regioni. I modelli di traffico passati non sono sempre un buon indicatore per stabilire se la quota di servizio è impostata correttamente.

Altrettanto importante è il fatto che il limite di nome della quota di servizio non è sempre lo stesso per ogni Regione. In una Regione il valore potrebbe essere cinque, in un'altra potrebbe essere dieci. La gestione di queste quote deve riguardare tutti gli stessi servizi, account e Regioni per garantire una resilienza costante sotto carico.

Riconciliare tutte le differenze di quota di servizio tra le diverse Regioni (Regione attiva o passiva) e creare processi per riconciliare continuamente queste differenze. I piani di test dei failover passivi delle Regioni sono raramente scalati in base alla capacità attiva di picco, il che significa che gli esercizi di game day o table top possono non riuscire a trovare le differenze nelle quote di servizio tra le Regioni e a mantenere i limiti corretti.

Deviazione delle quote di servizio, è molto importante da monitorare e valutare la condizione in cui i limiti delle quote di servizio per una specifica quota nominata vengono modificati in una Regione e non in tutte le Regioni. Si dovrebbe prendere in considerazione la possibilità di modificare la quota nelle Regioni con traffico o potenzialmente in grado di trasportare traffico.

- Seleziona gli account e le regioni pertinenti in base ai tuoi requisiti di servizio, di latenza, normativi e di ripristino di emergenza.
- Identifica le quote dei servizi per tutti gli account, le regioni e le zone di disponibilità pertinenti. Le restrizioni si riferiscono ad account e regione. Questi valori devono essere confrontati per far emergere le differenze.

Passaggi dell'implementazione

- Rivedi i valori Service Quotas che potrebbero aver superato il livello di rischio di utilizzo. AWS Trusted Advisor offre allarmi per la violazione di soglie dell'80% e del 90%.
- Rivedi i valori per le quote di servizio in qualsiasi Regione Passiva (in un progetto Attivo/Passivo). Verifica che il carico venga eseguito correttamente nelle Regioni secondarie in caso di guasto nella Regione primaria.

- Valuta automaticamente se si è verificata una deviazione delle quote di servizio tra le Regioni dello stesso account e agisci di conseguenza per modificare i limiti.
- Se le Unità Organizzative (UO) del cliente sono strutturate nel modo supportato, i modelli di quote di servizio devono essere aggiornati per riflettere le modifiche alle quote da applicare a più Regioni e account.
  - Crea un modello e associa le Regioni alla modifica della quota.
  - Rivedi tutti i modelli delle quote di servizio esistenti per qualsiasi modifica richiesta (Regione, limiti e account).

Risorse

Best practice correlate:

- REL01-BP01 Consapevolezza su quote e vincoli di servizio
- REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura
- REL01-BP04 Monitoraggio e gestione delle quote
- REL01-BP05 Automazione della gestione delle quote
- REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover
- REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL12-BP05 Test della resilienza tramite l'utilizzo dell'ingegneria del caos

Documenti correlati:

- Principio dell'affidabilità di AWS Well-Architected Framework: disponibilità
- AWS Service Quotas (precedentemente note come restrizioni dei servizi)
- Controlli delle best practice AWS Trusted Advisor (consulta la sezione Restrizioni dei servizi)
- Monitoraggio limite su AWS su risposte AWS
- Quote di servizio Amazon EC2
- Che cos'è Service Quotas?

- Come richiedere un aumento delle quote
- Endpoint e quote dei servizi
- Guida per l'utente di Service Quotas
- Monitoraggio delle quote per AWS
- Limiti di isolamento dei guasti di AWS
- Disponibilità con ridondanza
- AWS for Data
- In cosa consiste l'Integrazione continua?
- In cosa consiste la Distribuzione continua?
- Partner APN: partner per la gestione della configurazione
- Gestione del ciclo di vita dell'account in ambienti SaaS account-per-tenant su AWS
- Gestione e monitoraggio della limitazione delle API nei carichi di lavoro
- Visualizza i suggerimenti di AWS Trusted Advisor su scala con AWS Organizations
- Automazione dell'aumento dei limiti di servizio e supporto aziendale con AWS Control Tower

Video correlati:

- AWS Live re:Inforce 2019 - Service Quotas
- Visualizza e gestisci quote per i servizi AWS con Service Quotas
- Demo delle quote AWS IAM

Servizi correlati:

- Amazon CodeGuru Reviewer
- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK

- AWS Systems Manager
- Marketplace AWS

REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura

Identifica attentamente quote di servizio, vincoli del servizio e limiti delle risorse fisiche che non possono essere modificati. Progetta architetture per le applicazioni e i servizi in modo da impedire che questi limiti abbiano impatto sull'affidabilità.

Alcuni esempi includono la larghezza di banda di rete, le dimensioni di payload delle chiamate di funzioni serverless, il tasso di espansione dei limiti per un gateway API e le connessioni utente simultanee a un database.

Risultato desiderato: l'applicazione o il servizio ha le prestazioni previste in condizioni di traffico normale ed elevato. L'applicazione o il servizio è stato progettato per operare entro i limiti dei vincoli o delle quote di servizio fissi della risorsa.

Anti-pattern comuni:

- Scelta di una progettazione che usa una risorsa di un servizio, senza essere al corrente della presenza di vincoli che causeranno errori di progettazione durante il dimensionamento.
- Esecuzione di benchmark poco realistici e che raggiungono le quote di servizio fisse durante i test. Ad esempio, l'esecuzione di test a un limite di espansione per un periodo di tempo prolungato.
- Scelta di una progettazione che non può essere dimensionata o modificata in caso di superamento delle quote di servizio fisse. Ad esempio, dimensioni dei payload SQS di 256 KB.
- Mancata progettazione e implementazione della visibilità per monitorare e segnalare le soglie per le quote di servizio a rischio durante eventi di traffico elevato.

Vantaggi dell'adozione di questa best practice: possibilità di verificare che l'applicazione verrà eseguita a tutti i livelli di carico dei servizi previsti senza interruzioni o errori.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Diversamente dalle risorse e dalle quote di servizio flessibili che possono essere sostituite con unità di capacità maggiori, le quote di servizio fisse in AWS non possono essere modificate. Di conseguenza, tutti i servizi AWS di questo tipo devono essere valutati per identificare i possibili limiti fissi di capacità quando vengono usati per la progettazione di un'applicazione.

I limiti fissi vengono mostrati nella console Service Quotas. Se le colonne indicano `REGOLABILE = No`, il servizio ha un limite fisso. I limiti fissi vengono mostrati anche in alcune pagine di configurazione delle risorse. Ad esempio, per Lambda è previsto un limite fisso specifico che non può essere modificato.

Ad esempio, durante la progettazione di un'applicazione Python da eseguire in una funzione Lambda, l'applicazione deve essere valutata per determinare la probabilità che Lambda venga eseguito per più di 15 minuti. Se il codice potrebbe restare in esecuzione oltre questo limite della quota di servizio, devi prendere in considerazione tecnologie o progettazioni alternative. Se il limite viene raggiunto dopo l'implementazione nell'ambiente di produzione, l'applicazione sarà soggetta a errori o interruzioni finché non viene corretta. Diversamente dalle quote flessibili, non esiste alcun metodo per modificare i limiti, anche in caso di eventi di emergenza con livello di gravità 1.

Dopo aver implementato l'applicazione in un ambiente di test, è necessario adottare una strategia per determinare se vi sia la probabilità di raggiungere i limiti fissi. I test di stress, di carico e di chaos engineering devono fare parte del piano di test iniziale.

Passaggi dell'implementazione

- Esamina l'elenco completo dei servizi AWS che possono essere usati nella fase di progettazione dell'applicazione.
- Esamina i limiti di quota flessibili e fissi per tutti i servizi. Non tutti i limiti vengono indicati nella console Service Quotas. Alcuni servizi descrivono questi limiti in posizioni diverse.
- Nel progettare l'applicazione, esamina i principali fattori commerciali e tecnologici del carico di lavoro, come risultati aziendali, casi d'uso, sistemi dipendenti, obiettivi di disponibilità e oggetti di ripristino di emergenza. Fai in modo che siano questi fattori commerciali e tecnologici a orientare il processo di identificazione del sistema distribuito corretto per il carico di lavoro.
- Analizza il carico dei servizi tra regioni e account. Molti limiti fissi per i servizi variano a seconda della regione. Tuttavia, alcuni limiti dipendono dagli account.
- Analizza le architetture di resilienza per l'utilizzo delle risorse durante un guasto a livello di zona e di regione. Nel corso dello sviluppo di progettazioni multi-regione che usano approcci attivo/attivo, attivo/passivo con standby a caldo, attivo/passivo con standby a freddo e attivo/passivo con Pilot Light i casi di errore determineranno un utilizzo più elevato. Questo comportamento crea un possibile caso d'uso per il raggiungimento dei limiti fissi.

Risorse

Best practice correlate:

- REL01-BP01 Consapevolezza su quote e vincoli di servizio
- REL01-BP02 Gestione delle quote di servizio in più account e regioni
- REL01-BP04 Monitoraggio e gestione delle quote
- REL01-BP05 Automazione della gestione delle quote
- REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover
- REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL12-BP05 Test della resilienza tramite l'utilizzo dell'ingegneria del caos

Documenti correlati:

- Principio dell'affidabilità di AWS Well-Architected Framework: disponibilità
- AWS Service Quotas (precedentemente note come restrizioni dei servizi)
- Controlli delle best practice AWS Trusted Advisor (consulta la sezione Restrizioni dei servizi)
- Monitoraggio limite su AWS su risposte AWS
- Quote di servizio Amazon EC2
- Che cos'è Service Quotas?
- Come richiedere un aumento delle quote
- Endpoint e quote dei servizi
- Guida per l'utente di Service Quotas
- Monitoraggio delle quote per AWS
- Limiti di isolamento dei guasti di AWS
- Disponibilità con ridondanza
- AWS for Data
- In cosa consiste l'Integrazione continua?
- In cosa consiste la Distribuzione continua?
- Partner APN: partner per la gestione della configurazione
- Gestione del ciclo di vita dell'account in ambienti SaaS account-per-tenant su AWS

- Gestione e monitoraggio della limitazione delle API nei carichi di lavoro
- Visualizza i suggerimenti di AWS Trusted Advisor su scala con AWS Organizations
- Automazione dell'aumento dei limiti di servizio e supporto aziendale con AWS Control Tower
- Operazioni, risorse e chiavi di condizione per Service Quotas

Video correlati:

- AWS Live re:Inforce 2019 - Service Quotas
- Visualizza e gestisci quote per i servizi AWS con Service Quotas
- Demo delle quote AWS IAM
- AWS re:Invent 2018: Cicli chiusi e menti aperte: come assumere il controllo di sistemi grandi e piccoli

Strumenti correlati:

- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- Marketplace AWS

REL01-BP04 Monitoraggio e gestione delle quote

Valuta il tuo utilizzo potenziale e aumenta le quote in modo appropriato per una crescita pianificata dell'utilizzo.

Risultato desiderato: implementazione di sistemi attivi e automatici per la gestione e il monitoraggio. Queste soluzioni operative indicano che le soglie di utilizzo delle quote stanno per essere raggiunte. Questo problema può essere risolto in modo proattivo tramite modifiche alle quote richieste.

Anti-pattern comuni:

- Mancata configurazione del monitoraggio per verificare le soglie delle quote di servizio.
- Mancata configurazione del monitoraggio dei limiti fissi, anche se i valori non possono essere modificati.
- Valutazione errata della quantità di tempo necessaria per richiedere e ottenere la modifica di una quota flessibile, supponendo che sia immediata o rapida.
- Configurazione di allarmi per l'avvicinamento alle quote di servizio, ma senza alcun processo di risposta a un avviso.
- Configurazione di allarmi solo per i servizi supportati da AWS Service Quotas, senza monitorare altri servizi AWS.
- Valutazione errata della gestione delle quote per progettazioni di resilienza in più regioni, come gli approcci attivo/attivo, attivo/passivo con standby a caldo, attivo/passivo con standby a freddo e attivo/passivo con Pilot Light.
- Mancata valutazione delle differenze di quota tra regioni.
- Mancata valutazione delle esigenze in ogni regione per una richiesta di aumento di quota specifica.
- Mancato utilizzo di modelli per la gestione delle quote in più regioni.

Vantaggi dell'adozione di questa best practice: il monitoraggio automatico di AWS Service Quotas e il monitoraggio dell'utilizzo rispetto alle quote permettono di identificare l'avvicinamento a un limite di quota. Puoi usare questi dati di monitoraggio per limitare eventuali errori dovuti all'esaurimento della quota.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Per i servizi supportati, puoi monitorare le quote configurando servizi diversi in grado di eseguire una valutazione e quindi inviare avvisi o allarmi. In questo modo, il monitoraggio dell'utilizzo è più semplice e puoi ricevere avvisi all'avvicinamento delle quote. Gli allarmi possono essere attivati da AWS Config, funzioni Lambda, Amazon CloudWatch o AWS Trusted Advisor. Puoi anche usare filtri delle metriche in file di log CloudWatch per cercare ed estrarre modelli nei log, in modo da determinare se l'utilizzo si avvicina alle soglie delle quote.

Passaggi dell'implementazione

Per il monitoraggio:

- Acquisisci informazioni sull'attuale consumo di risorse, ad esempio bucket o istanze. Usa operazioni API dei servizi come l'API Amazon EC2 `DescribeInstances` per raccogliere informazioni sull'attuale consumo di risorse.
- Acquisisci le attuali quote essenziali e valide per i servizi usando:
  - AWS Service Quotas
  - AWS Trusted Advisor
  - Documentazione di AWS
  - Pagine specifiche dei servizi AWS
  - AWS Command Line Interface (AWS CLI)
  - AWS Cloud Development Kit (AWS CDK)
- Usa AWS Service Quotas, un servizio AWS che semplifica la gestione delle quote per oltre 250 servizi AWS da un'unica posizione.
- Usa i limiti del servizio Trusted Advisor per monitorare gli attuali limiti del servizio a soglie diverse.
- Usa la cronologia delle quote di servizio (console o AWS CLI) per verificare gli aumenti regionali.
- Confronta la modifica delle quote di servizio in ogni regione e ogni account per creare equivalenze, se necessario.

Per la gestione:

- Automatica: configura una regola AWS Config personalizzata per analizzare le quote di servizio tra regioni e confrontarle per individuare le differenze.
- Automatica: configura una regola Lambda personalizzata per analizzare le quote di servizio tra regioni e confrontarle per individuare le differenze.
- Manuale: analizza le quote di servizio tramite l'AWS CLI, l'API o la console AWS per esaminare le quote nelle diverse regioni e confrontarle per individuare le differenze. Segnala le differenze.
- Se vengono identificate differenze nelle quote tra regioni, richiedi una modifica della quota, se necessario.
- Esamina il risultato di tutte le richieste.

Risorse

Best practice correlate:

- REL01-BP01 Consapevolezza su quote e vincoli di servizio

- REL01-BP02 Gestione delle quote di servizio in più account e regioni
- REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura
- REL01-BP05 Automazione della gestione delle quote
- REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover
- REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL12-BP05 Test della resilienza tramite l'utilizzo dell'ingegneria del caos

Documenti correlati:

- Principio dell'affidabilità di AWS Well-Architected Framework: disponibilità
- AWS Service Quotas (precedentemente note come restrizioni dei servizi)
- Controlli delle best practice AWS Trusted Advisor (consulta la sezione Restrizioni dei servizi)
- Monitoraggio limite su AWS su risposte AWS
- Quote di servizio Amazon EC2
- Che cos'è Service Quotas?
- Come richiedere un aumento delle quote
- Endpoint e quote dei servizi
- Guida per l'utente di Service Quotas
- Monitoraggio delle quote per AWS
- Limiti di isolamento dei guasti di AWS
- Disponibilità con ridondanza
- AWS for Data
- In cosa consiste l'Integrazione continua?
- In cosa consiste la Distribuzione continua?
- Partner APN: partner per la gestione della configurazione
- Gestione del ciclo di vita dell'account in ambienti SaaS account-per-tenant su AWS

- Gestione e monitoraggio della limitazione delle API nei carichi di lavoro
- Visualizza i suggerimenti di AWS Trusted Advisor su scala con AWS Organizations
- Automazione dell'aumento dei limiti di servizio e supporto aziendale con AWS Control Tower
- Operazioni, risorse e chiavi di condizione per Service Quotas

Video correlati:

- AWS Live re:Inforce 2019 - Service Quotas
- Visualizza e gestisci quote per i servizi AWS con Service Quotas
- Demo delle quote AWS IAM
- AWS re:Invent 2018: Cicli chiusi e menti aperte: come assumere il controllo di sistemi grandi e piccoli

Strumenti correlati:

- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- Marketplace AWS

REL01-BP05 Automazione della gestione delle quote

Implementa strumenti per ricevere avvisi quando le soglie stanno per essere raggiunte. Puoi automatizzare le richieste di aumento delle quote utilizzando le API AWS Service Quotas.

Se integri il tuo database di gestione della configurazione (CMDB) o il sistema di ticketing con le Service Quotas, puoi automatizzare il monitoraggio delle richieste di aumento delle quote e delle

quote correnti. Oltre all'SDK AWS, Service Quotas offre automazione utilizzando AWS Command Line Interface (AWS CLI).

Anti-pattern comuni:

- Monitoraggio delle quote e dell'utilizzo nei fogli di calcolo.
- Esecuzione di report sull'utilizzo giornaliero, settimanale o mensile e successivo confronto dell'utilizzo con le quote.

Vantaggi dell'adozione di questa best practice: Il monitoraggio automatico delle quote di servizio AWS e il monitoraggio dell'utilizzo rispetto a tale quota ti consentiranno di sapere quando stai per raggiungere una quota. Puoi configurare l'automazione affinché ti aiuti a richiedere un aumento della quota quando necessario. Puoi decidere di ridurre alcune quote quando il tuo utilizzo tende alla direzione opposta per ottenere i vantaggi di riduzione del rischio (in caso di credenziali compromesse) e dei costi.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

- Impostazione del monitoraggio automatico: implementa strumenti utilizzando gli SDK per ricevere avvisi quando le soglie stanno per essere raggiunte.
  - Utilizza Service Quotas e potenzia il servizio con una soluzione di monitoraggio automatico delle quote come AWS Limit Monitor o un'offerta di Marketplace AWS.
    - What is Service Quotas? (Che cos'è Service Quotas?)
    - Monitoraggio delle quota su AWS – Soluzione AWS
  - Impostazione di risposte attivate in base alle soglie delle quote tramite l'utilizzo delle API di Amazon SNS e AWS Service Quotas.
  - Automazione dei test.
    - Configura le soglie delle restrizioni.
    - Integrazione con eventi di modifica di AWS Config, pipeline di implementazione, Amazon EventBridge o terze parti.
    - Imposta artificialmente soglie basse per le quote in modo da testare le risposte.
    - Configura i trigger per eseguire azioni adeguate in seguito alle notifiche e contatta AWS Support se necessario.
    - Attiva manualmente gli eventi di modifica.

- Esegui una giornata di gioco per testare il processo di modifica dell'aumento delle quote.

Risorse

Documenti correlati:

- Partner APN: partner per la gestione della configurazione
- Marketplace AWS: prodotti CMDB per il monitoraggio delle restrizioni
- AWS Service Quotas (precedentemente note come restrizioni dei servizi)
- Elenco di controllo delle best practice di AWS Trusted Advisor (consulta la sezione Restrizioni dei servizi)
- Monitoraggio delle quota su AWS – Soluzione AWS
- Quote di servizio di Amazon EC2
- What is Service Quotas? (Che cos'è Service Quotas?)

Video correlati:

- AWS Live re:Inforce 2019 - Service Quotas

REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover

Quando una risorsa restituisce un errore o è inaccessibile, può comunque essere conteggiata rispetto a una quota finché non viene terminata. Verifica che le quote tengano conto della sovrapposizione di risorse in errore o inaccessibili e della rispettiva sostituzione. Nel calcolare questo divario, devi considerare casi d'uso come errori di rete, regionali o delle zone di disponibilità.

Risultato desiderato: possibilità di gestire errori di piccola o grande entità relativi alle risorse o all'accessibilità delle risorse all'interno delle attuali soglie di servizio, tenendo conto degli errori delle zone, di rete o addirittura regionali nella pianificazione delle risorse.

Anti-pattern comuni:

- Impostazione delle quote di servizio in base alle esigenze attuali senza tenere conto degli scenari di failover.
- Calcolo della quota massima per un servizio senza tenere conto dei principali aspetti della stabilità statica.

- Calcolo della quota totale necessaria per ogni regione senza tenere conto delle potenziali risorse inaccessibili.
- Valutazione errata dei limiti di isolamento degli errori per alcuni servizi AWS e dei possibili modelli di utilizzo anomalo.

Vantaggi dell'adozione di questa best practice: quando eventi di interruzione dei servizi hanno impatto sulla disponibilità delle applicazioni, il cloud permette di implementare strategie per mitigare questi eventi o ripristinare i servizi. Queste strategie spesso includono la creazione di risorse aggiuntive per sostituire quelle in errore o inaccessibili. La strategia di gestione delle quote soddisferebbe queste condizioni di failover senza aggiungere altri fattori negativi dovuti al raggiungimento dei limiti dei servizi.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Nel valutare i limiti di quota, tieni conto dei casi di failover che possono verificarsi a causa di un peggioramento della situazione. È bene considerare i tipi di casi di failover seguenti:

- Un VPC interrotto o inaccessibile.
- Una sottorete inaccessibile.
- Una zona di disponibilità sufficientemente compromessa da avere impatto sull'accessibilità di molte risorse.
- Diverse route di rete o punti di ingresso e uscita bloccati o che sono stati modificati.
- Una regione sufficientemente compromessa da avere impatto sull'accessibilità di molte risorse.
- Presenza di più risorse, ma non tutte interessate da un errore in una regione o in una zona di disponibilità.

Errori come quelli elencati sopra possono essere il fattore scatenante dell'avvio di un evento di failover. La decisione relativa all'avvio del failover è unica per ogni situazione e cliente, in quanto l'impatto aziendale può variare notevolmente. Tuttavia, nel decidere operativamente l'avvio del failover dell'applicazione o dei servizi, la pianificazione della capacità delle risorse nella posizione di failover e delle quote correlate deve essere gestita prima dell'evento.

Esamina le quote per ogni servizio tenendo conto di possibili picchi più elevati del previsto. Questi picchi possono essere correlati a risorse ancora attive raggiungibili a causa di reti o autorizzazioni. Le risorse attive non terminate continuano a essere conteggiate rispetto al limite di quota del servizio.

Passaggi dell'implementazione

- Assicurati che vi sia una differenza sufficiente tra la quota di servizio e l'utilizzo massimo in modo da gestire un failover o la perdita di accessibilità.
- Determina le quote di servizio, specificando i pattern di implementazione, i requisiti di disponibilità e la crescita dei consumi.
- Richiedi aumenti delle quote, se necessario. Pianifica tenendo conto del tempo necessario affinché le richieste di aumento delle quote siano soddisfatte.
- Determina i requisiti di affidabilità, noti anche come numero di 9.
- Determina gli scenari di errore (ad esempio, perdita di un componente, una zona di disponibilità o una regione).
- Stabilisci la metodologia di implementazione (ad esempio, canary, blu/verde, rosso/nero o rolling).
- Includi un buffer appropriato (ad esempio, 15%) rispetto alla restrizione attuale.
- Includi calcoli per la stabilità statica (zonale e regionale) laddove appropriato.
- Pianifica la crescita dei consumi (ad esempio, monitora le tendenze dei consumi).
- Tieni conto dell'impatto della stabilità statica per i carichi di lavoro più critici. Valuta la conformità delle risorse a un sistema statisticamente stabile in tutte le regioni e le zone di disponibilità.
- Valuta se usare prenotazioni della capacità on demand per pianificare la capacità in anticipo rispetto a qualsiasi failover. Questa strategia può essere utile durante le pianificazioni aziendali più critiche per ridurre i possibili rischi legati all'ottenimento della quantità e del tipo di risorse corretti durante il failover.

Risorse

Best practice correlate:

- REL01-BP01 Consapevolezza su quote e vincoli di servizio
- REL01-BP02 Gestione delle quote di servizio in più account e regioni
- REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura
- REL01-BP04 Monitoraggio e gestione delle quote
- REL01-BP05 Automazione della gestione delle quote
- REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL12-BP05 Test della resilienza tramite l'utilizzo dell'ingegneria del caos

Documenti correlati:

- Principio dell'affidabilità di AWS Well-Architected Framework: disponibilità
- AWS Service Quotas (precedentemente note come restrizioni dei servizi)
- Controlli delle best practice AWS Trusted Advisor (consulta la sezione Restrizioni dei servizi)
- Monitoraggio limite su AWS su risposte AWS
- Quote di servizio Amazon EC2
- Che cos'è Service Quotas?
- Come richiedere un aumento delle quote
- Endpoint e quote dei servizi
- Guida per l'utente di Service Quotas
- Monitoraggio delle quote per AWS
- Limiti di isolamento dei guasti di AWS
- Disponibilità con ridondanza
- AWS for Data
- In cosa consiste l'Integrazione continua?
- In cosa consiste la Distribuzione continua?
- Partner APN: partner per la gestione della configurazione
- Gestione del ciclo di vita dell'account in ambienti SaaS account-per-tenant su AWS
- Gestione e monitoraggio della limitazione delle API nei carichi di lavoro
- Visualizza i suggerimenti di AWS Trusted Advisor su scala con AWS Organizations
- Automazione dell'aumento dei limiti di servizio e supporto aziendale con AWS Control Tower
- Operazioni, risorse e chiavi di condizione per Service Quotas

Video correlati:

- AWS Live re:Inforce 2019 - Service Quotas
- Visualizza e gestisci quote per i servizi AWS con Service Quotas

- Demo delle quote AWS IAM
- AWS re:Invent 2018: Cicli chiusi e menti aperte: come assumere il controllo di sistemi grandi e piccoli

Strumenti correlati:

- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- Marketplace AWS

REL 2. Come si pianifica la topologia di rete?

I carichi di lavoro sono spesso presenti in più ambienti. Questi includono più ambienti cloud (sia pubblicamente accessibili sia privati) e, possibilmente, l'infrastruttura del data center esistente. I piani devono includere considerazioni di rete, ad esempio connettività intrasistema e intersistema, gestione di indirizzi IP pubblici, gestione di indirizzi IP privati e risoluzione dei nomi di dominio.

Best practice

- REL02-BP01 Utilizzo di una connettività di rete a disponibilità elevata per gli endpoint pubblici del carico di lavoro
- REL02-BP02 Esecuzione del provisioning di connettività ridondante tra reti private nel cloud e negli ambienti on-premise.
- REL02-BP03 Verifica che l'allocazione delle sottoreti IP consenta l'espansione e la disponibilità:
- REL02-BP04 Preferire topologie hub-and-spoke rispetto a mesh da-molti-a-molti
- REL02-BP05 Applicazione di intervalli di indirizzi IP privati non sovrapposti in tutti gli spazi con indirizzi privati a cui sono connessi

REL02-BP01 Utilizzo di una connettività di rete a disponibilità elevata per gli endpoint pubblici del carico di lavoro

La creazione di connettività di rete a disponibilità elevata agli endpoint pubblici dei carichi di lavoro può ridurre i tempi di inattività dovuti a perdita di connettività e migliorare la disponibilità e il contratto sul livello di servizio del carico di lavoro. Per ottenere questo risultato, usa un servizio DNS a disponibilità elevata, reti di distribuzione di contenuti (CDN), API Gateway, bilanciamento del carico o proxy inversi.

Risultato desiderato: è essenziale pianificare, creare e rendere operativa una connettività di rete ad alta disponibilità per gli endpoint pubblici. Se il carico di lavoro diventa irraggiungibile a causa della perdita di connettività, il sistema apparirà ai clienti come arrestato, anche se il carico di lavoro è in esecuzione e disponibile. Combinando connettività di rete a disponibilità elevata e resiliente per gli endpoint pubblici del carico di lavoro, insieme a un'architettura resiliente per il carico di lavoro stesso, puoi offrire ai clienti la disponibilità e il livello di servizio migliori possibile.

AWS Global Accelerator, Amazon CloudFront, Amazon API Gateway, funzioni URL AWS Lambda, API AWS AppSync e Elastic Load Balancing (ELB) forniscono tutti endpoint pubblici a disponibilità elevata. Amazon Route 53 offre un servizio DNS altamente disponibile per la risoluzione dei nomi di dominio che permette di verificare che gli indirizzi degli endpoint pubblici possano essere risolti.

Puoi anche valutare applicazioni software Marketplace AWS per il bilanciamento del carico e l'esecuzione di proxy.

Anti-pattern comuni:

- Progettazione di un carico di lavoro a disponibilità elevata senza pianificare connettività DNS e di rete per la disponibilità elevata.
- Uso di indirizzi Internet pubblici su singoli container o istanze e gestione della connettività tramite DNS.
- Uso di indirizzi IP anziché nomi di dominio per l'individuazione dei servizi.
- Mancata esecuzione di test su scenari con perdita di connettività agli endpoint pubblici.
- Mancata analisi delle esigenze di velocità di trasmissione effettiva della rete e dei modelli di distribuzione.
- Nessuna attività di test e pianificazione per scenari di possibile interruzione della connettività di rete Internet agli endpoint pubblici del carico di lavoro.
- Distribuzione di contenuti (pagine Web, asset statici o file multimediali) in un'area geografica di grandi dimensioni senza usare una rete di distribuzione di contenuti.

- Nessuna pianificazione per la prevenzione di attacchi DDoS (Distributed Denial of Service). Gli attacchi DDoS rischiano di arrestare il traffico legittimo e di ridurre la disponibilità per gli utenti.

Vantaggi dell'adozione di questa best practice: la progettazione di connettività di rete altamente disponibile e resiliente garantisce che il carico di lavoro sia accessibile e disponibile per gli utenti.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

Alla base della creazione di connettività di rete a disponibilità elevata agli endpoint pubblici vi è l'instradamento del traffico. Per verificare che il traffico possa raggiungere gli endpoint, il servizio DNS deve essere in grado di risolvere i nomi di dominio negli indirizzi IP corrispondenti. Usa un sistema dei nomi di dominio (DNS) altamente disponibile e scalabile come Amazon Route 53 per gestire i record DNS del dominio. Puoi usare anche i controlli dell'integrità forniti da Amazon Route 53. I controlli dell'integrità verificano che l'applicazione sia raggiungibile, disponibile e funzionale e possono essere configurati in modo da simulare il comportamento degli utenti, come la richiesta di una pagina Web o un URL specifico. In caso di errore, Amazon Route 53 risponde alle richieste di risoluzione DNS e indirizza il traffico solo agli endpoint integri. Puoi anche valutare se usare le funzionalità di instradamento basato sulla latenza e GeoDNS offerte da Amazon Route 53.

Per verificare che il carico di lavoro stesso abbia disponibilità elevata, usa Elastic Load Balancing (ELB). Puoi usare Amazon Route 53 per indirizzare il traffico a ELB, che lo distribuisce alle istanze di calcolo di destinazione. Puoi anche usare Amazon API Gateway insieme a AWS Lambda per una soluzione serverless. I clienti possono anche eseguire carichi di lavoro in più Regioni AWS. Con il modello attivo/attivo multisito, il carico di lavoro può distribuire il traffico da più regioni. Con un modello attivo/passivo multisito, il carico di lavoro distribuisce il traffico dalla regione attiva, mentre i dati vengono replicati nella regione secondaria e diventano attivi in caso di errore nella regione primaria. Puoi usare i controlli dell'integrità in Route 53 per controllare il failover DNS da qualsiasi endpoint in una regione primaria a un endpoint in una regione secondaria, verificando che il carico di lavoro sia raggiungibile e disponibile per gli utenti.

Amazon CloudFront offre una semplice API per la distribuzione di contenuti con bassa latenza e velocità di trasferimento dati elevate gestendo le richieste tramite una rete di posizioni edge in tutto il mondo. Le reti di distribuzione di contenuti (CDN) operano per i clienti distribuendo i contenuti situati o memorizzati nella cache in una posizione vicina all'utente. In questo modo, la disponibilità dell'applicazione migliora, in quanto il carico del contenuto viene allontanato dai server verso posizioni edge di CloudFront. Le posizioni edge e le cache edge regionali includono copie

memorizzate nella cache del contenuto vicino agli utenti, per il recupero rapido e una raggiungibilità e una disponibilità maggiori del carico di lavoro.

Per i carichi di lavoro con utenti distribuiti in più aree geografiche, AWS Global Accelerator contribuisce a migliorare la disponibilità e le prestazioni delle applicazioni. AWS Global Accelerator fornisce indirizzi IP statici anycast che operano come punto di ingresso statico alle applicazioni ospitate in una o più Regioni AWS. In questo modo, il traffico può entrare nella rete globale AWS il più vicino possibile agli utenti, migliorando la raggiungibilità e la disponibilità del carico di lavoro. AWS Global Accelerator monitora anche l'integrità degli endpoint dell'applicazione usando controlli dell'integrità TCP, HTTP e HTTPS. Eventuali variazioni dell'integrità o della configurazione degli endpoint attivano il reindirizzamento del traffico degli utenti a endpoint integri che offrono le prestazioni e la disponibilità migliori agli utenti. Inoltre, AWS Global Accelerator ha una progettazione di isolamento degli errori che usa due indirizzi IPv4 statici gestiti da zone di rete indipendenti, migliorando la disponibilità delle applicazioni.

Per contribuire alla protezione dei clienti da attacchi DDoS, AWS offre AWS Shield Standard. Shield Standard è abilitato per impostazione predefinita e protegge da attacchi comuni contro l'infrastruttura (livelli 3 e 4), come i flood SYN/UDP e gli attacchi di riflessione, in modo da supportare la disponibilità elevata delle applicazioni in AWS. Per altre soluzioni di protezione da attacchi più sofisticati e di maggiore entità (come i flood UDP) e di tipo state-exhaustion (come i flood TCP SYN) e per proteggere le applicazioni in esecuzione su Amazon Elastic Compute Cloud (Amazon EC2), Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator e Route 53, puoi valutare se usare AWS Shield Advanced. Per la protezione da attacchi a livello di applicazione come i flood HTTP POST o GET, usa AWS WAF. AWS WAF può usare indirizzi IP, intestazioni HTTP, corpo HTTP, stringhe URI, iniezione SQL e condizioni di scripting cross-site per determinare se una richiesta debba essere bloccata o consentita.

Passaggi dell'implementazione

1. Configura un sistema DNS a disponibilità elevata: Amazon Route 53 è un servizio Web altamente disponibile e scalabile che opera come sistema dei nomi di dominio (DNS). Route 53 connette le richieste utente ad applicazioni Internet in esecuzione in AWS o on-premise. Per ulteriori informazioni, consulta Configurazione di Amazon Route 53 come servizio DNS.
2. Configura controlli dell'integrità: quando usi Route 53, verifica che solo le destinazioni integre siano risolvibili. Per iniziare, crea controlli dell'integrità in Route 53 e configura il failover DNS. Nel configurare controlli dell'integrità, è importante tenere conto degli aspetti seguenti:
   a. Modo in cui Amazon Route 53 determina se un controllo dell'integrità ha esito positivo
   b. Creazione, aggiornamento ed eliminazione di controlli dell'integrità

   c. Monitoraggio dello stato dei controlli dell'integrità e ricezione di notifiche
   d. Best practice per DNS in Amazon Route 53
3. Connessione del servizio DNS agli endpoint.
   a. Quando usi Elastic Load Balancing come destinazione per il traffico, usa Amazon Route 53 per creare un record alias che punti all'endpoint regionale del sistema di bilanciamento del carico. Durante la creazione del record alias, imposta l'opzione Valutazione dello stato target su Sì.
   b. Per carichi di lavoro serverless o API private con API Gateway, usa Route 53 per indirizzare il traffico ad API Gateway.
4. Opta per una rete di distribuzione di contenuti (CDN).
   a. Per distribuire contenuti usando posizioni edge più vicine all'utente, inizia acquisendo familiarità con il modo in cui CloudFront distribuisce contenuti.
   b. Inizia con una distribuzione semplice di CloudFront. CloudFront sa quindi determinare dove vuoi distribuire i contenuti e come monitorare e gestire la distribuzione di contenuti. Nel configurare la distribuzione di CloudFront, è importante tenere conto degli aspetti seguenti:
      i. Funzionamento della memorizzazione nella cache con posizioni edge CloudFront
      ii. Aumento della proporzione di richieste gestite direttamente dalle cache CloudFront (tasso di riscontri nella cache)
      iii. Uso di Amazon CloudFront Origin Shield
      iv. Ottimizzazione della disponibilità elevata con il failover delle origini in CloudFront
5. Configura la protezione a livello di applicazione: AWS WAF semplifica la protezione da exploit Web e bot comuni che possono compromettere la disponibilità e la sicurezza o consumare risorse eccessive. Per approfondire questi concetti, consulta Funzionamento di AWS WAF e prima di implementare protezioni da flood HTTP POST e GET a livello di applicazione, consulta Nozioni di base su AWS WAF. Puoi anche usare AWS WAF con CloudFront. Consulta la documentazione sul funzionamento di AWS WAF con funzionalità di Amazon CloudFront.
6. Configura protezione aggiuntiva da attacchi DDoS: per impostazione predefinita, tutti i clienti AWS ricevono protezione gratuita dagli attacchi DDoS comuni e più frequenti a livello di rete e di trasporto che prendono di mira il sito Web o l'applicazione con AWS Shield Standard. Per una protezione aggiuntiva delle applicazioni con connessione Internet su Amazon EC2, Elastic Load Balancing, Amazon CloudFront, AWS Global Accelerator e Amazon Route 53, puoi prendere in considerazione AWS Shield Advanced e consultare gli esempi di architetture resilienti ad attacchi DDoS. Per proteggere il carico di lavoro e gli endpoint pubblici da attacchi DDoS, consulta Nozioni di base su AWS Shield Advanced.

Risorse

Best practice correlate:

- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL10-BP02 Selezione delle posizioni appropriate per la tua implementazione multiposizione
- REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo durante il ripristino
- REL11-BP06 Invio di notifiche quando gli eventi influiscono sulla disponibilità

Documenti correlati:

- Partner APN: partner per la pianificazione della rete
- Marketplace AWS per l'infrastruttura di rete
- Che cos'è AWS Global Accelerator?
- Che cos'è Amazon CloudFront?
- Che cos'è Amazon Route 53?
- Che cos'è Elastic Load Balancing?
- Funzionalità di connettività di rete – Come definire gli aspetti di base del cloud
- Che cos'è Amazon API Gateway?
- Che cosa sono AWS WAF, AWS Shield e AWS Firewall Manager?
- Che cos'è il Sistema di controllo Amazon Route 53 per il ripristino di applicazioni?
- Configurazione di controlli dell'integrità personalizzati per il failover DNS

Video correlati:

- AWS re:Invent 2022: Miglioramento delle prestazioni e della disponibilità con AWS Global Accelerator
- AWS re:Invent 2020: Gestione del traffico globale con Amazon Route 53
- AWS re:Invent 2022: Esecuzione di applicazioni multi-AZ a disponibilità elevata
- AWS re:Invent 2022: Approfondimento dell'infrastruttura di rete AWS
- AWS re:Invent 2022: Creazione di reti resilienti

Esempi correlati:

- Ripristino di emergenza con il Sistema di controllo Amazon Route 53 per il ripristino di applicazioni (ARC)
- Workshop sull'affidabilità
- Workshop su AWS Global Accelerator

REL02-BP02 Esecuzione del provisioning di connettività ridondante tra reti private nel cloud e negli ambienti on-premise.

Implementa la ridondanza delle connessioni tra reti private nel cloud e negli ambienti on-premises per ottenere la resilienza della connettività. A tal fine, puoi implementare due o più collegamenti e percorsi di traffico, preservando la connettività in caso di errori di rete.

Anti-pattern comuni:

- Dipendi da una sola connessione di rete che crea un singolo punto di errore.
- Utilizzi un solo tunnel VPN o più tunnel che terminano nella stessa zona di disponibilità.
- Ti affidi a un solo ISP per la connettività VPN, suscettibile di guasto totale in caso di interruzione dell'ISP.
- Non implementi i protocolli di instradamento dinamico come BGP, fondamentali per reindirizzare il traffico durante le interruzioni di rete.
- Ignori i limiti di larghezza di banda dei tunnel VPN e sopravvaluti le capacità di backup.

Vantaggi dell'adozione di questa best practice: implementando una connettività ridondante tra il tuo ambiente cloud e l'ambiente aziendale/on-premises, i servizi dipendenti tra i due ambienti possono comunicare in maniera affidabile.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Quando si utilizza AWS Direct Connect per connettere la rete on-premises ad AWS, è possibile ottenere la massima resilienza di rete (SLA del 99,99%) impiegando connessioni separate che terminano su dispositivi diversi in più di una posizione on-premises e in più di una posizione AWS Direct Connect. Questa topologia offre resilienza agli errori dei dispositivi, ai problemi di connettività e alle interruzioni complete della posizione. In alternativa, puoi ottenere un'elevata resilienza (SLA del 99,9%) utilizzando due singole connessioni a più posizioni, con ciascuna posizione on-premises connessa a una singola posizione Direct Connect. Questo approccio offre protezione dalle

interruzioni della connettività causate da errori della fibra o guasti dei dispositivi e aiuta a mitigare le interruzioni complete della posizione. Il kit di strumenti di resilienza di AWS Direct Connect può aiutarti a progettare la tua topologia AWS Direct Connect.

Puoi anche considerare di utilizzare AWS Site-to-Site VPN che termina su AWS Transit Gateway come soluzione conveniente di backup sulla connessione primaria AWS Direct Connect. Questa configurazione abilita l'instradamento equal-cost multi-path (ECMP) su più tunnel VPN, consentendo un throughput fino a 50 Gbps, anche se ogni tunnel VPN è limitato a 1,25 Gbps. È importante notare, tuttavia, che AWS Direct Connect è ancora la scelta più efficace per ridurre al minimo le interruzioni di rete e fornire una connettività stabile.

Quando utilizzi le VPN su Internet per connettere l'ambiente cloud al tuo data center on-premises, configura due tunnel VPN come parte di un'unica connessione VPN sito-sito. Ogni tunnel deve terminare in una zona di disponibilità diversa per garantire l'alta disponibilità e utilizzare hardware ridondante per prevenire gli errori dei dispositivi on-premises. Inoltre, prendi in considerazione di utilizzare più connessioni Internet di vari provider di servizi Internet (ISP) per la tua posizione on-premises per evitare l'interruzione completa della connettività VPN dovuta al guasto di un singolo ISP. La scelta di ISP con instradamento e infrastrutture diversi, in particolare quelli con percorsi fisici separati verso gli endpoint AWS, offre un'elevata disponibilità della connettività.

Oltre alla ridondanza fisica con più connessioni AWS Direct Connect e più tunnel VPN, o una combinazione di entrambi, è fondamentale anche l'implementazione dell'instradamento dinamico del Border Gateway Protocol (BGP). Il BGP dinamico fornisce il reinstradamento automatico del traffico da un percorso all'altro in base alle condizioni della rete in tempo reale e alle policy configurate. Questo comportamento dinamico è particolarmente utile per mantenere la disponibilità della rete e la continuità del servizio in caso di errori di collegamento o rete. Seleziona rapidamente percorsi alternativi, migliorando la resilienza e l'affidabilità della rete.

Passaggi dell'implementazione

- Acquisisci la connettività ad alta disponibilità tra AWS e l'ambiente on-premises.
  - Utilizza più connessioni AWS Direct Connect o tunnel VPN tra reti private implementate separatamente.
  - Utilizza più posizioni AWS Direct Connect per l'alta disponibilità.
  - Se utilizzi più Regioni AWS, garantisci la ridondanza in almeno due di esse.
- Usa AWS Transit Gateway, quando possibile, per terminare la connessione VPN.

- Valuta le appliance di Marketplace AWS per terminare le VPN o estendere la SD-WAN ad AWS. Se utilizzi appliance di Marketplace AWS, distribuisci le istanze ridondanti per la disponibilità elevata in diverse zone di disponibilità.
- Fornisci una connessione ridondante all'ambiente on-premises.
  - Per soddisfare le esigenze di disponibilità, possono essere necessarie connessioni ridondanti a più Regioni AWS.
  - Utilizza il kit di strumenti di resilienza di AWS Direct Connect per iniziare.

Risorse

Documenti correlati:

- AWS Direct Connect Resiliency Recommendations
- Using Redundant Site-to-Site VPN Connections to Provide Failover
- Policy di routing e community BGP
- Active/Active and Active/Passive Configurations in AWS Direct Connect
- Partner APN: partner per la pianificazione della rete
- Marketplace AWS per l'infrastruttura di rete
- Amazon Virtual Private Cloud Connectivity Options Whitepaper
- Creazione di un'infrastruttura di rete AWS scalabile e sicura con più VPC
- Using redundant Site-to-Site VPN connections to provide failover
- Using the AWS Direct Connect Resiliency Toolkit to get started
- VPC Endpoints and VPC Endpoint Services (AWS PrivateLink)
- What is Amazon VPC?
- What is a transit gateway?
- What is AWS Site-to-Site VPN?
- Working with Direct Connect gateways

Video correlati:

- AWS re:Invent 2018: Advanced VPC Design and New Capabilities for Amazon VPC
- AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs

REL02-BP03 Verifica che l'allocazione delle sottoreti IP consenta l'espansione e la disponibilità:

Gli intervalli di indirizzi IP dei Amazon VPC devono essere sufficientemente ampi per soddisfare i requisiti del carico di lavoro, tenendo conto anche dell'espansione futura e dell'allocazione degli indirizzi IP alle sottoreti nelle zone di disponibilità. Sono inclusi sistemi di bilanciamento del carico, istanze EC2 e applicazioni basate su container.

Quando si pianifica la topologia di rete, il primo passo è definire lo spazio stesso degli indirizzi IP. Gli intervalli di indirizzi IP privati (secondo le linee guida RFC 1918) dovrebbero essere allocati per ogni VPC. Nell'ambito di questo processo, soddisfa i seguenti requisiti:

- Lascia spazi per indirizzi IP per più di un VPC per Regione.
- All'interno di un VPC, lascia spazio per più sottoreti affinché coprano più zone di disponibilità.
- Prendi in considerazione di lasciare spazio per un blocco CIDR inutilizzato all'interno di un VPC per un'espansione futura.
- Assicurati che sia disponibile spazio per gli indirizzi IP, al fine di soddisfare le esigenze di qualsiasi parco istanze Amazon EC2 transitorio che puoi utilizzare, ad esempio parchi istanze spot per il machine learning, cluster Amazon EMR o cluster Amazon Redshift. Una considerazione analoga andrebbe fatta per i cluster Kubernetes, come Amazon Elastic Kubernetes Service (Amazon EKS), poiché per impostazione predefinita a ciascun pod Kubernetes viene assegnato un indirizzo instradabile dal blocco CIDR VPC.
- Tieni presente che i primi quattro indirizzi IP e l'ultimo indirizzo IP in ogni blocco CIDR della sottorete sono riservati e non disponibili per l'uso.
- Tieni presente che il blocco CIDR VPC iniziale allocato al VPC non può essere modificato o eliminato, ma puoi aggiungere ulteriori blocchi CIDR non sovrapposti al VPC. I CIDR IPv4 della sottorete non possono essere modificati, mentre ciò è possibile con i CIDR IPv6.
- Il blocco CIDR VPC più grande possibile è /16 e il più piccolo è /28.
- Prendi in considerazione altre reti connesse (VPC, on-premises o altri provider cloud) e assicurati che lo spazio degli indirizzi IP non si sovrapponga. Per ulteriori informazioni, consulta REL02-BP05 Applicazione di intervalli di indirizzi IP privati non sovrapposti in tutti gli spazi con indirizzi privati a cui sono connessi.

Risultato desiderato: Una sottorete IP scalabile può aiutarti a far fronte alla crescita futura e a evitare inutili sprechi.

Anti-pattern comuni:

- Non prendere in considerazione la crescita futura, con conseguenti blocchi CIDR troppo piccoli e che richiedono una riconfigurazione, il che comporta tempi di inattività.
- Stima erronea del numero di indirizzi IP utilizzabili da un bilanciatore del carico.
- Distribuzione di numerosi sistemi di bilanciamento del carico a traffico elevato nelle stesse sottoreti.
- Utilizzo di meccanismi di dimensionamento automatico senza monitorare il consumo di indirizzi IP.
- Definizione di intervalli CIDR eccessivamente ampi ben oltre le aspettative di crescita futura, il che può portare a difficoltà di peering con altre reti con intervalli di indirizzi sovrapposti.

Vantaggi dell'adozione di questa best practice: In questo modo puoi consentire la crescita dei carichi di lavoro e continuare a fornire disponibilità man mano che incrementi le dimensioni.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Pianificazione della rete in base a crescita, compliance normativa e integrazione con altre reti. Senza una pianificazione adeguata, la crescita può essere sottovalutata, la compliance normativa può cambiare e l'implementazione di acquisizioni o di connessioni a reti private può rivelarsi difficile.

- Seleziona gli Account AWS e le Regioni pertinenti in base ai tuoi requisiti di servizio, di latenza, normativi e di ripristino di emergenza.
- Identifica le esigenze delle implementazioni di VPC regionali.
- Identifica le dimensioni dei VPC.
  - Stabilisci se intendi implementare connettività multi-VPC.
    - Che cos'è un Transit Gateway?
    - Connettività multi-VPC a singola Regione
  - Stabilisci se hai bisogno di reti separate a causa di requisiti normativi.
  - Crea VPC con blocchi CIDR di dimensioni adeguate per soddisfare le tue esigenze attuali e future.
    - Se non hai definito proiezioni di crescita, potresti preferire blocchi CIDR più grandi per ridurre il potenziale di riconfigurazione futura
  - Prendi in considerazione l'utilizzo di un indirizzo IPv6 per le sottoreti come parte di un VPC dual-stack. Un IPv6 è adatto per essere utilizzato in sottoreti private contenenti parchi istanze o contenitori temporanei che altrimenti richiederebbero un numero elevato di indirizzi IPv4.

Risorse

Best practice Well-Architected correlate:

- REL02-BP05 Applicazione di intervalli di indirizzi IP privati non sovrapposti in tutti gli spazi con indirizzi privati a cui sono connessi

Documenti correlati:

- Partner APN: partner per la pianificazione della rete
- Marketplace AWS per l'infrastruttura di rete
- Whitepaper: Opzioni di connettività di Amazon Virtual Private Cloud
- Connettività di rete di elevata disponibilità in più data center
- Connettività multi-VPC a singola Regione
- Che cos'è Amazon VPC?
- IPv6 su AWS
- IPv6 on reference architectures
- Amazon Elastic Kubernetes Service launches IPv6 support

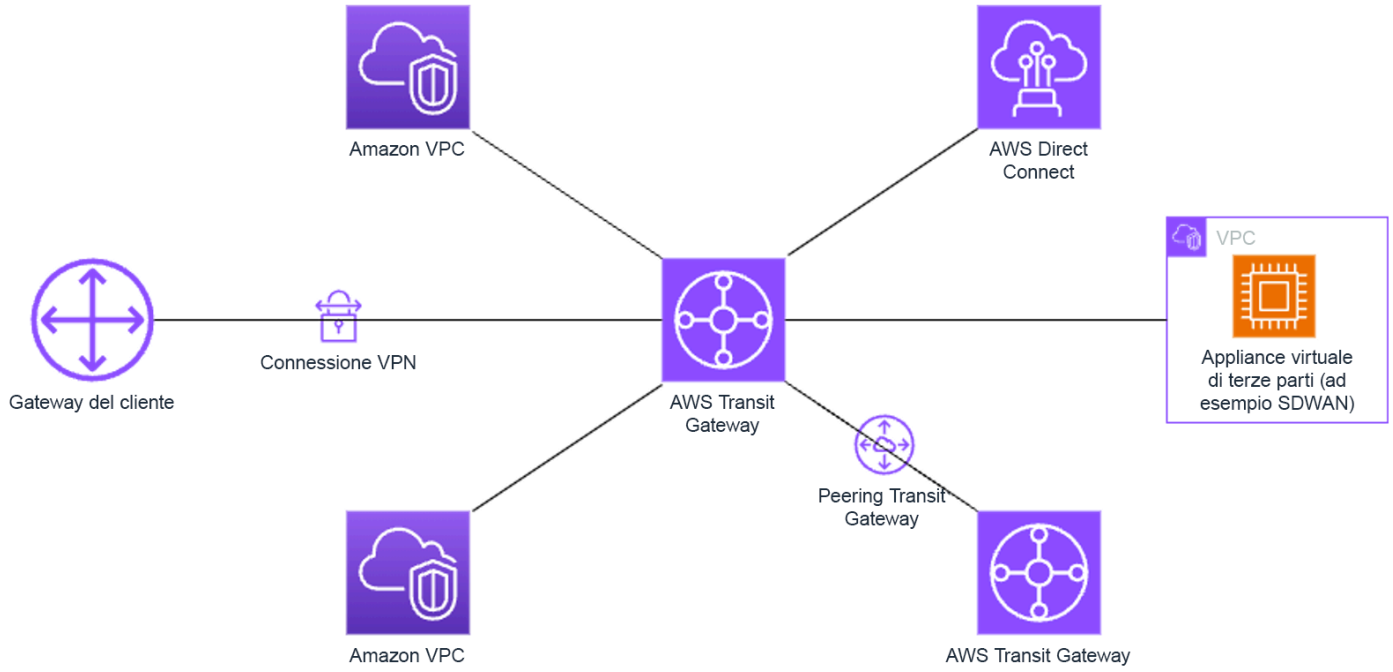
Video correlati:

- AWS re:Invent 2018: Advanced VPC Design and New Capabilities for Amazon VPC (Progettazione avanzata di VPC e nuove funzionalità per Amazon VPC) (NET303)
- AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs (NET406-R1)
- AWS re:Invent 2023: AWS Ready for what's next? Designing networks for growth and flexibility (NET310)

REL02-BP04 Preferire topologie hub-and-spoke rispetto a mesh da-molti-a-molti

Quando connetti più reti private, come cloud privati virtuali (VPC) e reti locali, è opportuno scegliere una topologia hub-and-spoke rispetto a una mesh. A differenza delle topologie mesh, in cui ogni rete si connette direttamente alle altre e aumenta la complessità e il sovraccarico di gestione, l'architettura hub-and-spoke centralizza le connessioni tramite un unico hub. Questa centralizzazione semplifica la struttura della rete e ne migliora il funzionamento, la scalabilità e il controllo.

AWS Transit Gateway è un servizio gestito, scalabile e a disponibilità elevata progettato per la creazione di reti hub-and-spoke su AWS. Serve da hub centrale della rete che fornisce la segmentazione, il routing centralizzato e la connessione semplificata agli ambienti cloud e on-premises. La figura seguente illustra come è possibile utilizzare AWS Transit Gateway per creare la topologia hub-and-spoke.



Anti-pattern comuni:

- Si complicano eccessivamente le policy di routing in un'architettura hub-and-spoke, riducendo l'efficienza della rete e ostacolando sia la risoluzione dei problemi sia la gestione proattiva.
- Una segmentazione insufficiente basata sul routing all'interno dell'hub potrebbe comportare vulnerabilità che potenzialmente espongono la rete ad accessi non autorizzati.
- Senza un'attenta ottimizzazione, il traffico instradato attraverso l'hub può comportare elevati costi di trasferimento dei dati, in particolare per il traffico che attraversa zone di disponibilità e regioni. L'uso di efficaci strategie di gestione del traffico è essenziale per controllare le spese.

Vantaggi dell'adozione di questa best practice: con l'aumento del numero di reti connesse, la gestione e l'espansione della connettività mesh diventano sempre più difficili. AWS Transit Gateway offre un hub gestito dimensionabile e affidabile per la creazione e il funzionamento delle topologie hub-and-

spoke. Con AWS Transit Gateway puoi stabilire connessioni e centralizzare il routing del traffico su più reti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Pianifica la rete.
- Crea un AWS Transit Gateway.
- Collega i VPC.
- Se necessario, crea connessioni VPN o gateway Direct Connect e associali a Transit Gateway.
- Definisci come viene instradato il traffico tra i VPC connessi e altre connessioni tramite la configurazione delle tabelle di routing di Transit Gateway.
- Usa Amazon CloudWatch per monitorare e modificare come necessario le configurazioni per l'ottimizzazione delle prestazioni e dei costi.

Risorse

Documenti correlati:

- What Is a Transit Gateway?
- Creazione di un'infrastruttura di rete AWS multi-VPC sicura e scalabile
- Building a global network using AWS Transit Gateway Inter-Region peering
- Amazon Virtual Private Cloud Connectivity Options
- APN Partner: partners that can help plan your networking
- Marketplace AWS for Network Infrastructure

Video correlati:

- AWS re:Invent 2023 - AWS networking foundations
- AWS re:Invent 2023 - Advanced VPC designs and new capabilities

REL02-BP05 Applicazione di intervalli di indirizzi IP privati non sovrapposti in tutti gli spazi con indirizzi privati a cui sono connessi

Gli intervalli di indirizzi IP di ogni VPC non devono sovrapporsi quando sono collegati in peering o connessi tramite Transit Gateway o VPN. Evita i conflitti di indirizzi IP tra VPC e ambienti on-premises o altri provider di servizi cloud utilizzati. Bisogna inoltre disporre di un modo per allocare gli intervalli di indirizzi IP privati quando necessario. Un sistema di gestione degli indirizzi IP (IPAM) può aiutarti ad automatizzare l'allocazione.

Risultato desiderato:

- Nessun conflitto di intervalli di indirizzi IP tra VPC, ambienti on-premises o altri provider di servizi cloud.
- La corretta gestione degli indirizzi IP consente di scalare più facilmente l'infrastruttura di rete per supportare la crescita e i cambiamenti dei requisiti di rete.

Anti-pattern comuni:

- Utilizzo nel VPC dello stesso intervallo di indirizzi IP usato on-premises, nella rete aziendale o in altro provider di servizi cloud.
- Non tenere traccia degli intervalli IP dei VPC utilizzati per distribuire i carichi di lavoro.
- Ricorso a processi manuali di gestione degli indirizzi IP, come i fogli di calcolo.
- Utilizzo di blocchi CIDR sovradimensionati o sottodimensionati, con conseguente spreco di indirizzi IP o spazio di indirizzi insufficiente per il carico di lavoro.

Vantaggi derivanti dall'adozione di questa best practice: la pianificazione attiva della rete garantisce di non avere più occorrenze dello stesso indirizzo IP nelle reti interconnesse. In questo modo si evitano problemi di instradamento in parti del carico di lavoro che utilizzano le diverse applicazioni.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Utilizza un sistema IPAM, ad esempio Amazon VPC IP Address Manager, per monitorare e gestire l'uso del CIDR. Su Marketplace AWS sono disponibili anche diversi IPAM. Valuta il tuo utilizzo potenziale su AWS, aggiungi intervalli CIDR ai VPC esistenti e crea i VPC per consentire la crescita pianificata dell'utilizzo.

Passaggi dell'implementazione

- Acquisisci il consumo attuale del CIDR, ad esempio VPC e sottoreti.
  - Utilizza le operazioni delle API di servizi per raccogliere il consumo attuale di CIDR.
  - Usa Amazon VPC IP Address Manager per individuare le risorse.
- Acquisisci l'utilizzo attuale delle sottoreti.
  - Utilizza le operazioni delle API di servizio per raccogliere le sottoreti per ogni VPC di ciascuna regione.
  - Usa Amazon VPC IP Address Manager per individuare le risorse.
- Registra l'uso attuale.
- Verifica se hai creato intervalli di indirizzi IP sovrapposti.
- Calcola la capacità inutilizzata.
- Individua gli intervalli di indirizzi IP sovrapposti. Puoi eseguire la migrazione a un nuovo intervallo di indirizzi o prendere in considerazione l'utilizzo di tecniche quali il gateway NAT privato o AWS PrivateLink, se hai l'esigenza di connettere intervalli sovrapposti.

Risorse

Best practice correlate:

- Protezione delle reti

Documenti correlati:

- Partner APN: partner per la pianificazione della rete
- Marketplace AWS per l'infrastruttura di rete
- Amazon Virtual Private Cloud Connectivity Options Whitepaper
- Connettività di rete di elevata disponibilità in più data center
- Connecting Networks with Overlapping IP Ranges
- What is Amazon VPC?
- Che cos'è IPAM?

Video correlati:

- AWS re:Invent 2023 - Advanced VPC designs and new capabilities
- AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs
- AWS re:Invent 2023 - Ready for what's next? Designing networks for growth and flexibility
- AWS re:Invent 2021 - {New Launch} Manage your IP addresses at scale on AWS

Architettura del carico di lavoro

Domande

- REL 3. Come si progetta l'architettura del servizio di carico di lavoro?
- REL 4. Come si progettano le interazioni in un sistema distribuito per evitare errori?
- REL 5. Come si progettano le interazioni in un sistema distribuito per mitigare o affrontare gli errori?

REL 3. Come si progetta l'architettura del servizio di carico di lavoro?

Creazione di carichi di lavoro altamente scalabili e affidabili utilizzando un'architettura orientata ai servizi (SOA) o un'architettura di microservizi. L'architettura orientata ai servizi (SOA) è la pratica di rendere i componenti software riutilizzabili tramite interfacce di servizio. L'architettura dei microservizi va oltre, per rendere i componenti più piccoli e semplici.

Best practice

- REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro
- REL03-BP02 Creazione di servizi focalizzati su domini e funzionalità aziendali specifici
- REL03-BP03 Fornitura di contratti di servizio per API

REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro

La segmentazione del carico di lavoro è importante quando vengono determinati i requisiti di resilienza dell'applicazione. L'architettura monolitica deve essere evitata se possibile. Valuta invece con particolare attenzione quali componenti dell'applicazione possono essere suddivisi in microservizi. A seconda dei requisiti dell'applicazione, ciò potrebbe risultare in una combinazione di architettura orientata ai servizi (SOA) e microservizi, laddove possibile. I carichi di lavoro stateless sono maggiormente idonei a essere implementati come microservizi.

Risultato desiderato: i carichi di lavoro devono essere supportabili, scalabili e devono essere caratterizzati dalla minore interdipendenza possibile.

Quando scegli come segmentare il carico di lavoro, trova il giusto compromesso tra i vantaggi e le complessità. Ciò che è giusto per un nuovo prodotto al primo lancio è diverso dai requisiti di un carico di lavoro creato per ridimensionare le risorse. Durante la rifattorizzazione (riprogettazione) di un monolito, dovrai considerare la capacità dell'applicazione di supportare la suddivisione in servizi stateless. La suddivisione dei servizi in elementi più piccoli consente a team ristretti e ben definiti di svilupparli e gestirli. Tuttavia, servizi di piccole dimensioni possono introdurre complessità, che includono un eventuale aumento della latenza, un debug più complesso e un maggiore carico operativo.

Anti-pattern comuni:

- Il microservizio Death Star rappresenta una situazione in cui i componenti atomici diventano così interdipendenti che un errore verificatosi in un componente genera un errore molto più grande, rendendo i componenti rigidi e fragili se considerati come monolito.

Vantaggi dell'adozione di questa best practice:

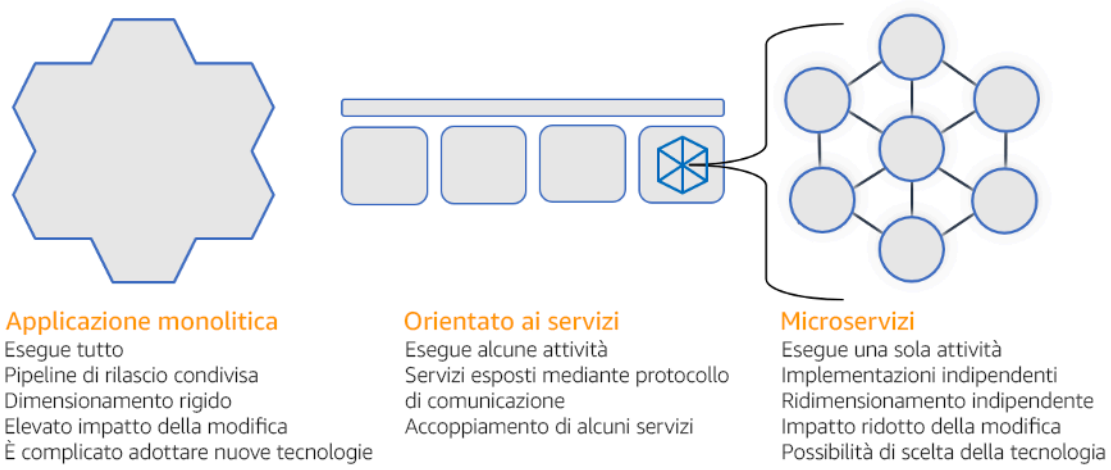
- Segmenti più specifici comportano maggiore agilità, flessibilità organizzativa e scalabilità.
- Riduzione dell'impatto derivante dall'interruzione dei servizi.
- I componenti dell'applicazione possono avere requisiti di disponibilità diversi, che a loro volta possono essere supportati da una segmentazione più atomica.
- Responsabilità ben definite per i team che supportano il carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Scegli il tipo di architettura in base al tipo di segmentazione del carico di lavoro. Scegli una SOA o un'architettura di microservizi (o, in alcuni rari casi, un'architettura monolitica). Anche se scegli di iniziare con un'architettura monolitica, devi assicurarti che sia modulare e possa evolvere in SOA o microservizi man mano che il prodotto si dimensiona con l'adozione da parte degli utenti. La SOA e i microservizi offrono rispettivamente una segmentazione più piccola, preferita come architettura moderna scalabile e affidabile, ma ci sono compromessi da considerare soprattutto quando si distribuisce un'architettura di microservizi.

Uno dei principali compromessi è che ora disponi di un'architettura di calcolo distribuita che può rendere più difficile il raggiungimento dei requisiti di latenza degli utenti ed è presente un'ulteriore complessità nel debug e nel tracciamento delle interazioni degli utenti. Puoi utilizzare AWS X-Ray per risolvere questo problema. Un altro effetto da considerare è l'aumento della complessità operativa man mano che aumenta il numero di applicazioni che gestisci, che richiede la distribuzione di più componenti di indipendenza.



Architettura monolitica, orientata ai servizi e di microservizi

Passaggi dell'implementazione

- Determina l'architettura più appropriata per rifattorizzare (riprogettare) o creare l'applicazione. SOA e microservizi offrono segmentazione rispettivamente di dimensioni minori, preferita in quanto architettura moderna, scalabile e affidabile. SOA può essere un buon compromesso per ottenere una segmentazione di dimensioni minori, evitando al contempo alcune delle complessità dei microservizi. Per ulteriori dettagli, consulta I compromessi dei microservizi.
- Se il carico di lavoro è adatto e la tua organizzazione può supportarla, è consigliabile utilizzare un'architettura di microservizi per ottenere la massima agilità e affidabilità. Per ulteriori dettagli, consulta Implementing Microservices on AWS (Implementazione di microservizi in AWS).
- Considera l'ipotesi di attenerti al modello Strangler Fig per eseguire la rifattorizzazione (riprogettazione) di un monolito in componenti più piccoli. Ciò comporta la graduale sostituzione di componenti specifici dell'applicazione con nuove applicazioni e nuovi servizi. AWS Migration Hub Refactor Spaces funge da punto di partenza per la rifattorizzazione incrementale. Per ulteriori dettagli, consulta Seamlessly migrate on-premises legacy workloads using a strangler pattern (Migrazione senza problemi di carichi di lavoro legacy on-premise mediante un modello Strangler).

- L'implementazione di microservizi può richiedere un meccanismo di individuazione dei servizi per consentire ai servizi distribuiti di comunicare tra loro. AWS App Mesh può essere utilizzato con architetture orientate ai servizi per offrire rilevamento e accesso affidabili ai servizi. AWS Cloud Map può inoltre essere utilizzato per il rilevamento dinamico dei servizi basato su DNS.
- In caso di migrazione da un monolito a una SOA, Amazon MQ può aiutare a colmare il divario come bus del servizio durante la riprogettazione delle applicazioni legacy nel cloud.
- Per i monoliti esistenti con un unico database condiviso, scegli come riorganizzare i dati in segmenti più piccoli. Questa riorganizzazione può avvenire per unità aziendale, schema di accesso o struttura dei dati. A questo punto del processo di rifattorizzazione (riprogettazione), deve orientare la scelta verso un database di tipo relazionale o non relazionale (NoSQL). Per ulteriori dettagli, consulta From SQL to NoSQL (Da SQL a NoSQL).

Livello di impegno per il piano di implementazione: alto

Risorse

Best practice correlate:

- REL03-BP02 Creazione di servizi focalizzati su domini e funzionalità aziendali specifici

Documenti correlati:

- Amazon API Gateway: configurazione di una REST API mediante OpenAPI
- Cosa si intende per architettura orientata ai servizi?
- Bounded Context (un modello centrale in Domain-Driven Design)
- Implementazione di microservizi in AWS
- I compromessi dei microservizi
- Microservizi: una definizione di questo nuovo termine di architettura
- Implementazione di microservizi in AWS
- What is AWS App Mesh? (Che cos'è AWS App Mesh?)

Esempi correlati:

- Iterative App Modernization Workshop (Workshop sulla modernizzazione delle applicazioni interattive)

Video correlati:

- Delivering Excellence with Microservices on AWS (Implementazione dell'eccellenza con i microservizi in AWS)

REL03-BP02 Creazione di servizi focalizzati su domini e funzionalità aziendali specifici

L'architettura orientata ai servizi (SOA) definisce servizi con funzioni ben delineate determinate dalle esigenze aziendali. I microservizi utilizzano modelli di dominio e contesto delimitato per tracciare i limiti dei servizi lungo i confini del contesto aziendale. Concentrarsi sui domini e sulle funzionalità aziendali aiuta i team a definire requisiti di affidabilità indipendenti per i propri servizi. I contesti delimitati isolano e incapsulano la logica aziendale, consentendo ai team di ragionare meglio su come gestire gli errori.

Risultato desiderato: ingegneri e parti interessate aziendali definiscono congiuntamente contesti delimitati e li utilizzano per progettare sistemi come servizi che soddisfano funzioni aziendali specifiche. Questi team utilizzano pratiche consolidate come l'event storming per definire i requisiti. Le nuove applicazioni sono concepite come servizi con confini ben definiti e con accoppiamento debole. I monoliti esistenti vengono scomposti in contesti delimitati e la progettazione dei sistemi si sposta verso architetture SOA o microservizi. Quando i monoliti vengono rifattorizzati, vengono applicati approcci consolidati come contesti a bolle e schemi di decomposizione dei monoliti.

I servizi orientati al dominio vengono eseguiti come uno o più processi che non condividono lo stato. Rispondono in modo indipendente alle fluttuazioni della domanda e gestiscono gli scenari di errore alla luce dei requisiti specifici del dominio.

Anti-pattern comuni:

- I team sono formati su domini tecnici specifici come UI e UX, middleware o database anziché su domini aziendali specifici.
- Le applicazioni coprono le responsabilità di dominio. I servizi che coprono contesti delimitati possono essere più difficili da gestire, richiedere maggiori sforzi di test ed esigere la partecipazione di più team di dominio agli aggiornamenti software.
- Le dipendenze a livello di dominio, come le librerie di entità di dominio, sono condivise tra i servizi, in modo che le modifiche per il dominio di un servizio richiedano modifiche ad altri domini dei servizi.

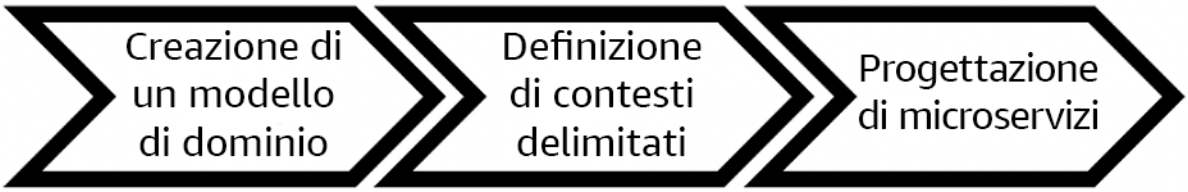
- I contratti di servizio e la logica aziendale non esprimono le entità in un linguaggio di dominio comune e coerente, con il risultato di livelli di traduzione che complicano i sistemi e aumentano le attività di debug.

Vantaggi dell'adozione di questa best practice: le applicazioni sono progettate come servizi indipendenti limitati da domini aziendali e utilizzano un linguaggio aziendale comune. I servizi sono testabili e implementabili in modo indipendente. I servizi soddisfano i requisiti di resilienza specifici del dominio implementato.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

La decisione basata sul dominio (DDD) costituisce l'approccio fondamentale alla progettazione e alla creazione di software attorno ai domini aziendali. È utile utilizzare un framework esistente quando si creano servizi incentrati sui domini aziendali. Quando si utilizzano applicazioni monolitiche esistenti, è possibile sfruttare i modelli di decomposizione che forniscono tecniche consolidate per modernizzare le applicazioni in servizi.



Decisione basata sul dominio

Passaggi dell'implementazione

- I team possono organizzare eventi di event storming per identificare rapidamente eventi, comandi, aggregati e domini.
- Una volta che le entità e le funzioni di dominio sono state definite in un contesto di dominio, puoi dividere il tuo dominio in servizi utilizzando il contesto delimitato, dove le entità che condividono caratteristiche e attributi simili vengono raggruppate insieme. Con il modello diviso in contesti, emerge un modello su come delimitare i microservizi.
  - Ad esempio, le entità del sito Web Amazon.com possono includere elementi quali pacchetti, distribuzione, pianificazione, prezzo, sconto e valuta.

    - Il pacchetto, la distribuzione e la pianificazione sono raggruppati nel contesto di spedizione, mentre il prezzo, lo sconto e la valuta sono raggruppati nel contesto dei prezzi.
- Scomporre i monoliti in microservizi delinea i modelli per la rifattorizzazione dei microservizi. L'utilizzo di modelli per la decomposizione in base a capacità aziendale, sottodominio o transazione si allinea bene agli approcci basati sul dominio.
- Tecniche di strategia come il contesto a bolle consentono di introdurre la decisione basata sul dominio (DDD) in applicazioni esistenti o precedenti senza riscritture anticipate e impegni completi nei confronti di DDD. In un approccio basato sul contesto a bolle, viene stabilito un contesto delimitato utilizzando una mappatura e un coordinamento dei servizi, oppure il livello anti-danneggiamento (ACL), che protegge il modello di dominio appena definito dalle influenze esterne.

Dopo aver eseguito l'analisi del dominio e definito le entità e i contratti di servizio, i team possono utilizzare i servizi AWS per implementare la progettazione basata sul dominio come servizi basati sul cloud.

- Inizia a sviluppare definendo test che applichino le regole aziendali del tuo dominio. Lo sviluppo basato sui test (TDD) e lo sviluppo basato sul comportamento (BDD) aiutano i team a focalizzare i servizi sulla risoluzione dei problemi aziendali.
- Seleziona i servizi AWS che soddisfano al meglio i requisiti del tuo dominio aziendale e l' architettura di microservizi:
    - AWS Serverless consente al team di concentrarsi su una logica di dominio specifica anziché sulla gestione di server e infrastrutture.
    - I container in AWS semplificano la gestione della tua infrastruttura, in modo da poterti concentrare sui requisiti del tuo dominio.
    - I database dedicati ti aiutano ad adattare i requisiti del tuo dominio al tipo di database più idoneo.
- La creazione di architetture esagonali su AWS delinea un framework per integrare la logica aziendale nei servizi che funzionano a ritroso da un dominio aziendale per soddisfare i requisiti funzionali e, quindi, per collegare adattatori di integrazione. I modelli che separano i dettagli dell'interfaccia dalla logica aziendale con i servizi AWS aiutano i team a concentrarsi sulla funzionalità del dominio e a migliorare la qualità del software.

Risorse

Best practice correlate:

- REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro
- REL03-BP03 Fornitura di contratti di servizio per API

Documenti correlati:

- Microservizi AWS
- Implementazione di microservizi in AWS
- How to break a Monolith into Microservices (Come trasformare un monolite in microservizi)
- Getting Started with DDD when Surrounded by Legacy Systems (Iniziare con il DDD quando si è circondati da sistemi legacy)
- Domain-Driven Design: Tackling Complexity in the Heart of Software (Progettazione basata sul dominio: affrontare la complessità al cuore del software)
- La creazione di architetture esagonali su AWS
- Scomporre i monoliti in microservizi
- Event Storming
- Messaggi tra contesti limitati
- Microservizi
- Sviluppo basato su test
- Sviluppo basato sul comportamento

Esempi correlati:

- Workshop sul cloud aziendale nativo
- Progettazione di microservizi cloud nativi su AWS (da DDD/EventStormingWorkshop)

Strumenti correlati:

- Database su Cloud AWS
- Serverless su AWS
- Container in AWS

REL03-BP03 Fornitura di contratti di servizio per API

I contratti di assistenza sono accordi documentati tra produttori di API e utenti definiti in una definizione di API leggibile dal computer. Una strategia di controllo delle versioni dei contratti consente agli utenti di continuare a utilizzare l'API esistente e migrare le applicazioni a un'API più recente quando sono pronte. L'implementazione da parte del produttore può avvenire in qualsiasi momento, purché il processo sia conforme al contratto. Il team dei servizi può utilizzare lo stack tecnologico scelto per soddisfare il contratto API.

Risultato desiderato:

Anti-pattern comuni: Le applicazioni realizzate con architetture orientate ai servizi o con architetture di microservizi sono in grado di funzionare in modo indipendente pur essendo caratterizzate da una dipendenza dal runtime integrata. Le modifiche apportate a un utente o produttore di API non pregiudicano la stabilità dell'intero sistema quando entrambe le parti sono conformi a un contratto API comune. I componenti che comunicano tramite le API di servizio possono eseguire release funzionali indipendenti, aggiornamenti delle dipendenze di runtime o eseguire il failover su un sito di ripristino di emergenza con un impatto reciproco minimo o nullo. Inoltre, i servizi discreti sono in grado di eseguire il dimensionamento in modo indipendente assorbendo la richiesta di risorse senza che gli altri servizi debbano ridimensionarsi di conseguenza.

- Creazione di API di servizio senza schemi fortemente tipizzati. Ciò si traduce in API che non possono essere utilizzate per generare collegamenti API e payload che non possono essere convalidati a livello di codice.
- Non adottare una strategia di controllo delle versioni, che costringa gli utenti delle API all'aggiornamento e rilascio o all'esito negativo dell'operazione al variare dei contratti di servizio.
- Messaggi di errore che divulgano dettagli sull'implementazione del servizio sottostante anziché descrivere errori di integrazione nel contesto e nel linguaggio del dominio.
- Non utilizzare contratti API per sviluppare casi di test e simulare implementazioni API per consentire test indipendenti dei componenti del servizio.

Vantaggi dell'adozione di questa best practice: i sistemi distribuiti composti da componenti che comunicano tramite contratti di servizio API possono migliorare l'affidabilità. Gli sviluppatori possono rilevare potenziali problemi nelle prime fasi del processo di sviluppo con il controllo del tipo durante la compilazione per verificare che le richieste e le risposte siano conformi al contratto API e che i campi obbligatori siano presenti. I contratti API forniscono una chiara interfaccia di

documentazione automatica per le API e garantiscono una migliore interoperabilità tra sistemi e linguaggi di programmazione diversi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Dopo aver individuato i domini aziendali e determinato la segmentazione del carico di lavoro, puoi sviluppare le API dei tuoi servizi. Innanzitutto, definisci contratti di servizio leggibili dal computer per le API, quindi implementa una strategia di controllo delle versioni delle API. Quando sei pronto per integrare servizi su protocolli comuni come REST, GraphQL o eventi asincroni, puoi incorporare servizi AWS nell'architettura per integrare i componenti con contratti API fortemente tipizzati.

I servizi AWS per i contratti API di servizio

includono servizi AWS come Amazon API Gateway, AWS AppSynce Amazon EventBridge nell'architettura per utilizzare i contratti di servizio API nell'applicazione. Amazon API Gateway è un valido supporto per l'integrazione con i servizi AWS direttamente nativi e altri servizi Web. API Gateway supporta la specifica OpenAPI e il controllo delle versioni. AWS AppSync è un endpoint gestito da GraphQL configurato definendo uno schema GraphQL per definire un'interfaccia di servizio per query, mutazioni e sottoscrizioni. Amazon EventBridge utilizza schemi di eventi per definire eventi e generare associazioni di codice per gli eventi.

Passaggi dell'implementazione

- Definisci innanzitutto un contratto per la tua API. Un contratto esprimerà le capacità di un'API e definirà oggetti e campi di dati fortemente tipizzati per l'input e l'output dell'API.
- Quando configuri le API in API Gateway, puoi importare ed esportare le specifiche OpenAPI per gli endpoint.
  - L'importazione di una definizione OpenAPI semplifica la creazione dell'API e può essere integrata con l'infrastruttura AWS come strumenti di codice come AWS Serverless Application Model e AWS Cloud Development Kit (AWS CDK).
  - L'esportazione di una definizione API semplifica l'integrazione con gli strumenti di test delle API e fornisce agli utenti di servizi una specifica di integrazione.
- Puoi definire e gestire le API GraphQL con AWS AppSync mediante la definizione di un file di schema GraphQL per generare l'interfaccia del contratto e semplificare l'interazione con modelli REST complessi, più tabelle di database o servizi legacy.

- I progetti AWS Amplify integrati con AWS AppSync generano file di query JavaScript fortemente tipizzati da utilizzare nell'applicazione, nonché una libreria client GraphQL AWS AppSync per le tabelle Amazon DynamoDB .
- Quando si utilizzano eventi di servizio da Amazon EventBridge, gli eventi sono conformi agli schemi già esistenti nel registro degli schemi o definiti con la specifica OpenAPI. Con uno schema definito nel registro, puoi anche generare associazioni client dal contratto dello schema per integrare il codice con gli eventi.
- Estensione o definizione della versione dell'API. L'estensione di un'API è un'opzione più semplice quando si aggiungono campi che possono essere configurati con campi facoltativi o valori predefiniti per i campi obbligatori.
  - I contratti basati su JSON per protocolli come REST e GraphQL possono essere adatti per l'estensione del contratto.
  - I contratti basati su XML per protocolli come SOAP devono essere testati con gli utenti dei servizi per determinare se l'estensione del contratto è possibile.
- Quando esegui il controllo delle versioni di un'API, valuta la possibilità di implementare il controllo delle versioni proxy laddove un lato viene usato per supportare le versioni in modo che la logica possa essere gestita in un'unica base di codice.
  - Con API Gateway puoi usare mappature di richieste e risposte per semplificare l'inclusione delle modifiche del contratto stabilendo un lato per fornire valori predefiniti per i nuovi campi o per eliminare i campi rimossi da una richiesta o una risposta. Con questo approccio, il servizio sottostante può avere un'unica base di codice.

Risorse

Best practice correlate:

- REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro
- REL03-BP02 Creazione di servizi focalizzati su domini e funzionalità aziendali specifici
- REL04-BP02 Implementazione di dipendenze "loosely coupled"
- REL05-BP03 Controllo e limitazione delle chiamate di ripetizione
- REL05-BP05 Impostazione dei timeout dei client

Documenti correlati:

- Cos'è un'interfaccia di programmazione dell'applicazione (API)?

- Implementazione di microservizi in AWS
- I compromessi dei microservizi
- Microservizi: una definizione di questo nuovo termine di architettura
- Microservizi in AWS
- Utilizzo delle estensioni API Gateway di OpenAPI
- Specifica OpenAPI
- GraphQL: schemi e tipi
- Associazioni di codice Amazon EventBridge

Esempi correlati:

- Amazon API Gateway: configurazione di una REST API mediante OpenAPI
- Applicazione CRUD da Amazon API Gatewaya Amazon DynamoDB utilizzando OpenAPI
- Modelli di integrazione di applicazioni moderne in un'era serverless: integrazione dei servizi API Gateway
- Implementazione del controllo delle versioni API Gateway basato sull'intestazione con Amazon CloudFront
- AWS AppSync: creazione di un'applicazione client

Video correlati:

- Utilizzo di OpenAPI AWS SAM per la gestione di API Gateway

Strumenti correlati:

- Amazon API Gateway
- AWS AppSync
- Amazon EventBridge

REL 4. Come si progettano le interazioni in un sistema distribuito per evitare errori?

I sistemi distribuiti si basano sulle reti di comunicazione per interconnettere i componenti (ad esempio server o servizi). Il carico di lavoro deve funzionare in modo affidabile nonostante la perdita o la latenza dei dati in queste reti. I componenti del sistema distribuito devono funzionare in modo da non

influire negativamente su altri componenti o sul carico di lavoro. Queste best practice prevengono gli errori e migliorano il tempo medio tra errori (MTBF).

Best practice

- REL04-BP01 Identificazione del tipo di sistema distribuito da cui si dipende
- REL04-BP02 Implementazione di dipendenze "loosely coupled"
- REL04-BP03 Esecuzione di un lavoro costante
- REL04-BP04 Rendere tutte le risposte idempotenti

REL04-BP01 Identificazione del tipo di sistema distribuito da cui si dipende

I sistemi distribuiti possono essere sincroni, asincroni o batch. I sistemi sincroni devono elaborare le richieste il più rapidamente possibile e comunicare tra loro effettuando chiamate di richiesta e risposta sincrone utilizzando i protocolli HTTP/S, REST o RPC (Remote Procedure Call). I sistemi asincroni comunicano tra loro scambiando i dati in modo asincrono tramite un servizio intermediario senza associare singoli sistemi. I sistemi batch ricevono un grande volume di dati di input, eseguono i processi di dati automatizzati senza intervento umano e generano i dati di output.

Risultato desiderato: progetta un carico di lavoro che interagisca efficacemente con le dipendenze sincrone, asincrone e batch.

Anti-pattern comuni:

- Il carico di lavoro attende a tempo indeterminato una risposta dalle dipendenze, con eventuale timeout del client del carico di lavoro, senza informazioni sulla ricezione della richiesta.
- Il carico di lavoro utilizza una catena di sistemi dipendenti che si chiamano tra loro in modo sincrono. A tal fine, ogni sistema deve essere disponibile ed elaborare correttamente la richiesta prima che l'intera catena possa essere completata, con conseguenti comportamenti e disponibilità complessiva potenzialmente fragili.
- Il carico di lavoro comunica con le dipendenze in modo asincrono e si basa sul concetto di distribuzione garantita dei messaggi esattamente una volta, quando spesso è ancora possibile ricevere messaggi duplicati.
- Il carico di lavoro non utilizza strumenti di pianificazione batch adeguati e consente l'esecuzione simultanea dello stesso processo batch.

Vantaggi dell'adozione di questa best practice: è comune che un determinato carico di lavoro implementi uno o più stili di comunicazione tra sincrona, asincrona e batch. Questa best practice consente di identificare i diversi compromessi associati a ogni stile di comunicazione per rendere il carico di lavoro in grado di tollerare interruzioni in tutte le sue dipendenze.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Le sezioni seguenti contengono le linee guida per l'implementazione generali e specifiche di ogni tipo di dipendenza.

Informazioni generali

- Assicurati che gli obiettivi del livello di servizio (SLO) di prestazioni e affidabilità offerti dalle dipendenze soddisfino i requisiti di prestazioni e affidabilità del tuo carico di lavoro.
- Utilizza i servizi di osservabilità AWS per monitorare i tempi di risposta e i tassi di errore e assicurarti che la dipendenza fornisca un servizio ai livelli necessari per il tuo carico di lavoro.
- Individua le potenziali sfide che il carico di lavoro può affrontare quando comunica con le dipendenze. I sistemi distribuiti presentano un'ampia serie di sfide che possono aumentare la complessità dell'architettura, gli oneri operativi e i costi. Le sfide più comuni includono latenza, interruzioni della rete, perdita dei dati, dimensionamento e ritardo nella replica dei dati.
- Implementa un solido sistema di gestione e registrazione degli errori per aiutarti a risolvere i problemi quando la dipendenza restituisce errori.

Dipendenza sincrona

Nelle comunicazioni sincrone, il carico di lavoro invia una richiesta alla dipendenza e blocca l'operazione in attesa della risposta. Quando la dipendenza riceve la richiesta, cerca di gestirla il prima possibile e invia una risposta al carico di lavoro. Una sfida significativa con la comunicazione sincrona è rappresentata dall'accoppiamento temporale, che richiede che il carico di lavoro e le sue dipendenze siano disponibili nello stesso momento. Quando il carico di lavoro deve comunicare in modo sincrono con le dipendenze, valuta le seguenti linee guida:

- Il carico di lavoro non deve fare affidamento su più dipendenze sincrone per eseguire una singola funzione. Questa catena di dipendenze aumenta la fragilità complessiva perché tutte le dipendenze nel percorso devono essere disponibili affinché la richiesta venga completata correttamente.

- Quando una dipendenza non è integra o non è disponibile, applica le strategie di gestione degli errori e riprova. Evita di usare un comportamento bimodale. Il comportamento bimodale si verifica quando il carico di lavoro presenta un comportamento diverso in modalità normale e in modalità di guasto. Per maggiori dettagli sul comportamento bimodale, consulta REL11-BP05 Utilizzo della stabilità statica per evitare un comportamento bimodale.
- Tieni presente che anticipare l'errore (fail fast) è meglio che far aspettare il carico di lavoro. Ad esempio, in AWS Lambda Developer Guide viene descritto come gestire i tentativi e gli errori quando si richiamano le funzioni Lambda.
- Imposta i timeout per quando il carico di lavoro chiama la dipendenza. Questa tecnica evita di aspettare troppo a lungo o all'infinito una risposta. Per una spiegazione utile di questo problema, consulta Tuning AWS Java SDK HTTP request settings for latency-aware Amazon DynamoDB applications.
- Riduci al minimo il numero di chiamate effettuate dal carico di lavoro alla dipendenza per soddisfare una singola richiesta. Le lunghe chiamate aumentano l'associazione e la latenza.

Dipendenza asincrona

Per disaccoppiare temporaneamente il carico di lavoro dalla dipendenza, è necessario che comunichino in modo asincrono. Con l'approccio asincrono, il carico di lavoro può continuare qualsiasi altra elaborazione senza dover attendere che la dipendenza o la catena di dipendenze invii la risposta.

Quando il carico di lavoro deve comunicare in modo asincrono con la dipendenza, tieni conto delle seguenti indicazioni:

- Determina in base al caso d'uso e ai requisiti se utilizzare la messaggistica o lo streaming di eventi. La messaggistica consente al carico di lavoro di comunicare con la dipendenza inviando e ricevendo messaggi tramite un broker di messaggi. Lo streaming di eventi consente al carico di lavoro e alle dipendenze di utilizzare un servizio di streaming per pubblicare e sottoscrivere gli eventi, distribuiti come flussi di dati continui che devono essere elaborati il prima possibile.

- La messaggistica e lo streaming di eventi gestiscono i messaggi in modo diverso, quindi devi stabilire i compromessi in base a:
  - Priorità dei messaggi: i broker di messaggi possono elaborare i messaggi ad alta priorità prima dei messaggi normali. Nello streaming di eventi, tutti i messaggi hanno la stessa priorità.

  - Consumo di messaggi: i broker di messaggi assicurano che gli utenti ricevano il messaggio. Gli utenti che utilizzano lo streaming di eventi devono tenere traccia dell'ultimo messaggio letto.
  - Ordinamento di messaggi: con la messaggistica non è garantita la ricezione dei messaggi nell'ordine esatto in cui vengono inviati, a meno che non si utilizzi un approccio first in-first-out (FIFO). Lo streaming di eventi mantiene sempre l'ordine in cui i dati sono stati prodotti.
  - Eliminazione di messaggi: con la messaggistica, l'utente deve eliminare il messaggio dopo averlo elaborato. Il servizio di streaming di eventi aggiunge il messaggio a un flusso e lo conserva fino alla scadenza del periodo di conservazione del messaggio. Questa policy di eliminazione rende lo streaming di eventi adatto alla riproduzione dei messaggi.
- Definisci in che modo il carico di lavoro riconosce il completamento del lavoro della dipendenza. Ad esempio, quando il carico di lavoro richiama una funzione Lambda in modo asincrono, Lambda inserisce l'evento in una coda e restituisce una risposta positiva senza informazioni aggiuntive. Una volta completata l'elaborazione, la funzione Lambda può inviare il risultato a una destinazione, configurabile in base all'esito positivo o negativo.
- Crea il tuo carico di lavoro per gestire i messaggi duplicati utilizzando l'idempotenza. Con l'idempotenza i risultati del carico di lavoro non cambiano anche se il carico di lavoro viene generato più volte per lo stesso messaggio. È importante considerare che i servizi di messaggistica o streaming recapitano nuovamente il messaggio se si verifica un errore di rete o se non è stata ricevuta la conferma.
- Se il carico di lavoro non riceve una risposta dalla dipendenza, deve inviare nuovamente la richiesta. Valuta la possibilità di limitare il numero di tentativi per preservare la CPU, la memoria e le risorse di rete del carico di lavoro al fine di gestire le altre richieste. Nella documentazione di AWS Lambda viene indicato come gestire gli errori per l'invocazione asincrona.
- Utilizza gli strumenti di osservabilità, debug e monitoraggio adeguati per gestire e usare la comunicazione asincrona del carico di lavoro con le relative dipendenze. Puoi utilizzare Amazon CloudWatch per monitorare i servizi di messaggistica e streaming di eventi. Puoi anche ottimizzare il carico di lavoro con AWS X-Ray per ottenere rapidamente approfondimenti per la risoluzione dei problemi.

Dipendenza batch

I sistemi batch acquisiscono i dati di input, avviano una serie di processi per elaborarli e producono i dati di output, senza intervento manuale. A seconda delle dimensioni dei dati, i processi possono durare da minuti a diversi giorni in alcuni casi. Quando il carico di lavoro comunica con la dipendenza batch, tieni conto delle seguenti indicazioni:

- Definisci la finestra temporale in cui il carico di lavoro deve eseguire il processo batch. Puoi impostare un modello di ricorrenza per il carico di lavoro per richiamare il sistema batch, ad esempio ogni ora o alla fine di ogni mese.
- Determina la posizione dei dati di input e di output elaborati. Scegli un servizio di archiviazione, ad esempio Amazon Simple Storage Services (Amazon S3), Amazon Elastic File System (Amazon EFS) e Amazon FSx for Lustre, per consentire al carico di lavoro di leggere e scrivere file su larga scala.
- Se il carico di lavoro deve richiamare più processi batch, puoi usare AWS Step Functions per semplificare l'orchestrazione dei processi batch eseguiti in AWS oppure on-premises. Questo progetto di esempio dimostra l'orchestrazione di processi batch utilizzando Step Functions, AWS Batch e Lambda.
- Monitora i processi batch per individuare eventuali anomalie, ad esempio un processo che richiede più tempo del dovuto per essere completato. Puoi utilizzare strumenti come CloudWatch Container Insights per monitorare ambienti e processi AWS Batch. In tal caso, il carico di lavoro interrompe l'inizio del processo successivo e comunica l'eccezione al team competente.

Risorse

Documenti correlati:

- Cloud AWS Operations: Monitoraggio e osservabilità
- Amazon Builders' Library: Difficoltà dei sistemi distribuiti
- REL11-BP05 Utilizzo della stabilità statica per evitare un comportamento bimodale
- AWS Lambda Developer Guide: Error handling and automatic retries in AWS Lambda
- Tuning AWS Java SDK HTTP request settings for latency-aware Amazon DynamoDB applications
- Messaggistica in AWS
- Cosa sono i flussi di dati?
- AWS Lambda Developer Guide: Asynchronous invocation
- Domande frequenti su Amazon Simple Queue Service: Code FIFO
- Amazon Kinesis Data Streams Developer Guide: Handling Duplicate Records
- Amazon Simple Queue Service Developer Guide: Available CloudWatch metrics for Amazon SQS
- Amazon Kinesis Data Streams Developer Guide: Monitoring the Amazon Kinesis Data Streams Service with Amazon CloudWatch

- AWS X-Ray Developer Guide: AWS X-Ray concepts
- AWS Samples on GitHub: AWS Step functions Complex Orchestrator App
- AWS Batch User Guide: AWS Batch CloudWatch Container Insights

Video correlati:

- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS (COP310)

Strumenti correlati:

- Amazon CloudWatch
- Amazon CloudWatch Logs
- AWS X-Ray
- Amazon Simple Storage Services (Amazon S3)
- Amazon Elastic File System (Amazon EFS)
- Amazon FSx for Lustre
- AWS Step Functions
- AWS Batch

REL04-BP02 Implementazione di dipendenze "loosely coupled"

Le dipendenze come sistemi di accodamento, sistemi di streaming, flussi di lavoro e sistemi di bilanciamento del carico sono "loosely coupled" (con accoppiamento debole). L'accoppiamento debole aiuta a isolare il comportamento di un componente dagli altri componenti che dipendono da esso, aumentando la resilienza e l'agilità.

Nei sistemi con accoppiamento stretto, le modifiche a un componente possono richiedere modifiche agli altri componenti basati su di esso, con conseguente riduzione delle prestazioni di tutti i componenti. L'accoppiamento debole interrompe questa dipendenza, in modo che i componenti dipendenti debbano conoscere solo l'interfaccia con versione e pubblicata. L'implementazione di un accoppiamento debole tra dipendenze isola un errore all'interno di una dipendenza affinché non influenzi l'altra.

L'accoppiamento debole consente di modificare il codice o aggiungere funzionalità a un componente riducendo al minimo il rischio per gli altri componenti che dipendono da esso. Consente inoltre una

resilienza granulare a livello di componente in cui è possibile impiegare la scalabilità orizzontale o persino modificare l'implementazione sottostante della dipendenza.

Per migliorare ulteriormente la resilienza tramite accoppiamento debole, rendi le interazioni dei componenti asincrone laddove possibile. Questo modello è idoneo a qualsiasi interazione che non richieda una risposta immediata e laddove la conferma della registrazione di una richiesta sia sufficiente. Include un componente che genera eventi e un altro che li utilizza. I due componenti non si integrano tramite un'interazione diretta point-to-point, ma in genere attraverso un livello di archiviazione intermedio durevole, come una coda Amazon SQS o una piattaforma di dati in streaming come Amazon Kinesis o AWS Step Functions.

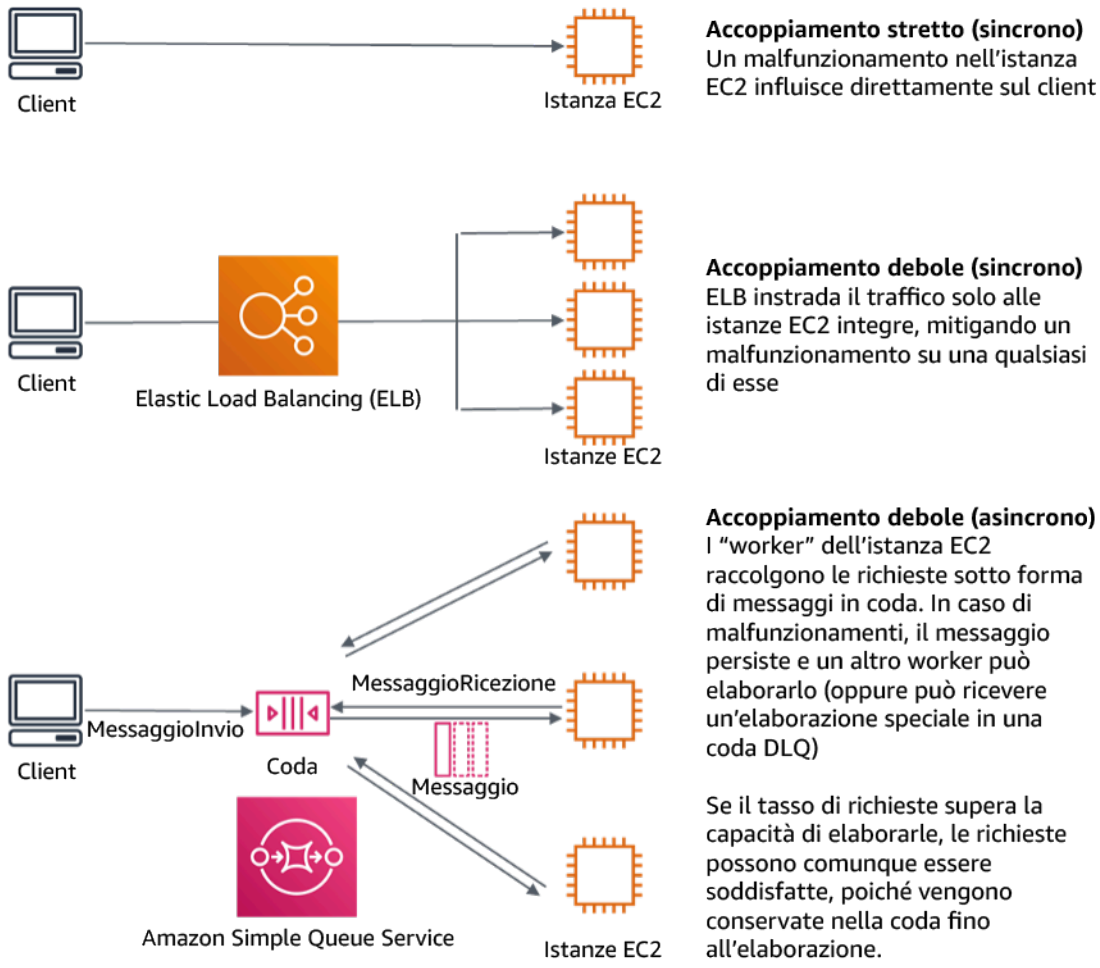


Figura 4. Le dipendenze come i sistemi di accodamento e i sistemi di bilanciamento del carico sono "loosely coupled"

Le code Amazon SQS ed Elastic Load Balancer sono solo due modi per aggiungere un livello intermedio per l'accoppiamento debole. Le architetture basate su eventi possono anche essere create in Cloud AWS utilizzando Amazon EventBridge, che può astrarre i client (produttori di eventi)

dai servizi a cui fanno affidamento (consumatori di eventi). Amazon Simple Notification Service (Amazon SNS) è una soluzione efficace quando hai bisogno di messaggistica da-molti-a-molti, dalla velocità di trasmissione effettiva elevata e basata su push. Utilizzando gli argomenti di Amazon SNS, i sistemi di pubblicazione possono inviare messaggi a un numero elevato di endpoint sottoscrittori per l'elaborazione parallela.

Mentre le code offrono diversi vantaggi, nella maggior parte dei sistemi hard real-time, le richieste più vecchie di una soglia temporale (spesso secondi) dovrebbero essere considerate obsolete (il client ha abbandonato e non è più in attesa di una risposta) e non elaborate. In questo modo, è possibile elaborare invece le richieste più recenti (e probabilmente ancora valide).

Risultato desiderato: l'implementazione di dipendenze con accoppiamento debole consente di ridurre al minimo l'area esposta ai guasti a livello di componente e ciò aiuta a diagnosticare e risolvere i problemi. Inoltre, semplifica i cicli di sviluppo, consentendo ai team di implementare le modifiche a livello modulare senza pregiudicare le prestazioni di altri componenti che dipendono da esso. Questo approccio offre la possibilità di impiegare la scalabilità orizzontale a livello di componente in base al fabbisogno di risorse, nonché di utilizzare un componente che contribuisce alla competitività in termini di costi.

Anti-pattern comuni:

- Implementazione di un carico di lavoro monolitico.
- Invocazione diretta di API tra livelli di carico di lavoro senza funzionalità di failover o elaborazione asincrona della richiesta.
- Accoppiamento stretto utilizzando dati condivisi. I sistemi con accoppiamento debole dovrebbero evitare di condividere i dati tramite database condivisi o altre forme di archiviazione dei dati con accoppiamento stretto, che possono reintrodurre l'accoppiamento stretto e compromettere la scalabilità.
- Ignorare la contropressione. Il carico di lavoro dovrebbe essere in grado di rallentare o arrestare i dati in arrivo quando un componente non è in grado di elaborarli alla stessa velocità.

Vantaggi dell'adozione di questa best practice: l'accoppiamento debole aiuta a isolare il comportamento di un componente dagli altri componenti che dipendono da esso, aumentando la resilienza e l'agilità. L'errore in un componente è isolato dagli altri.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

Implementazione di dipendenze "loosely coupled". Esistono varie soluzioni che consentono di creare applicazioni con accoppiamento debole. Queste includono, ad esempio, servizi per l'implementazione di code completamente gestite, flussi di lavoro automatizzati, reazione agli eventi e API, che possono aiutare a isolare il comportamento dei componenti dagli altri componenti e, di conseguenza, aumentare la resilienza e l'agilità.

- Crea architetture basate su eventi: Amazon EventBridge ti aiuta a creare architetture basate sugli eventi con accoppiamento debole e distribuite.
- Implementazione di code nei sistemi distribuiti: è possibile utilizzare Amazon Simple Queue Service (Amazon SQS) per integrare e disaccoppiare i sistemi distribuiti.
- Containerizza i componenti come microservizi: i microservizi consentono ai team di creare applicazioni composte da piccoli componenti indipendenti che comunicano tramite API ben definite. Amazon Elastic Container Service (Amazon ECS) e Amazon Elastic Kubernetes Service (Amazon EKS) possono aiutarti a iniziare a utilizzare più rapidamente i container.
- Gestisci i flussi di lavoro con Step Functions: Step Functions semplifica il coordinamento di più servizi AWS in flussi di lavoro flessibili.
- Sfrutta le architetture di messaggistica publish-subscribe (pub/sub): Amazon Simple Notification Service (Amazon SNS) fornisce il servizio di consegna dei messaggi dagli editori agli abbonati (noti anche come produttori e consumatori).

Passaggi dell'implementazione

- I componenti in un'architettura basata su eventi vengono avviati dagli eventi. Gli eventi sono azioni che si verificano in un sistema, ad esempio un utente che aggiunge un articolo a un carrello. Quando un'azione ha successo, viene generato un evento che attiva il successivo componente del sistema.
  - Creazione di applicazioni basate su eventi con Amazon EventBridge
  - AWS re:Invent 2022 - Designing Event-Driven Integrations using Amazon EventBridge
- I sistemi di messaggistica distribuiti sono composti da tre parti principali che devono essere implementate per un'architettura basata su code. Includono i componenti del sistema distribuito, la coda utilizzata per il disaccoppiamento (distribuita su server Amazon SQS) e i messaggi in coda. Un sistema tipico prevede produttori che inviano il messaggio alla coda e il consumatore che riceve il messaggio dalla coda. La coda archivia i messaggi su più server Amazon SQS per garantire la ridondanza.

  - Architettura Amazon SQS di base
  - Invia messaggi tra applicazioni distribuite con Amazon Simple Queue Service
- I microservizi, se ben utilizzati, migliorano la manutenibilità e aumentano la scalabilità, poiché i componenti ad accoppiamento debole sono gestiti da team indipendenti. Consentono inoltre l'isolamento dei comportamenti in un unico componente in caso di modifiche.
  - Implementazione di microservizi in AWS
  - Let's Architect! Architecting microservices with containers
- Con AWS Step Functions è possibile eseguire moltissime operazioni, ad esempio creare applicazioni distribuite, automatizzare i processi e orchestrare microservizi. L'orchestrazione di più componenti in un flusso di lavoro automatizzato consente di disaccoppiare le dipendenze nell'applicazione.
  - Create a Serverless Workflow with AWS Step Functions and AWS Lambda
  - Nozioni di base su AWS Step Functions

Risorse

Documenti correlati:

- Amazon EC2: Ensuring Idempotency
- The Amazon Builders' Library: Difficoltà dei sistemi distribuiti
- The Amazon Builders' Library: Reliability, constant work, and a good cup of coffee
- What is Amazon EventBridge?
- What is Amazon Simple Queue Service?
- Break up with your monolith
- Orchestrate Queue-based Microservices with AWS Step Functions and Amazon SQS
- Architettura Amazon SQS di base
- Architettura basata su code

Video correlati:

- AWS New York Summit 2019: Intro to Event-driven Architectures and Amazon EventBridge (MAD205)
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small ARC337 (includes loose coupling, constant work, static stability)

- AWS re:Invent 2019: Moving to event-driven architectures (SVS308)
- AWS re:Invent 2019: Scalable serverless event-driven applications using Amazon SQS and Lambda (API304)
- AWS re:Invent 2019: Scalable serverless event-driven applications using Amazon SQS and Lambda
- AWS re:Invent 2022 - Designing event-driven integrations using Amazon EventBridge
- AWS re:Invent 2017: Elastic Load Balancing Deep Dive and Best Practices

REL04-BP03 Esecuzione di un lavoro costante

I sistemi possono fallire quando si verificano modifiche rapide e di grandi dimensioni nel carico. Ad esempio, se il carico di lavoro effettua un controllo dell'integrità di migliaia di server deve inviare ogni volta lo stesso payload delle dimensioni (uno snapshot completo dello stato corrente). Indipendentemente dal fatto che non ci siano server guasti, o che lo siano tutti, il sistema di controllo dello stato esegue un lavoro costante con modifiche rapide e di piccole dimensioni.

Ad esempio, se il sistema di controllo dello stato monitora 100.000 server, il carico su di esso è nominale al di sotto del tasso di errore normalmente basso del server. Tuttavia, se un evento importante rendesse la metà di questi server non integra, il sistema di controllo dello stato sarebbe sovraccarico nel tentativo di aggiornare i sistemi di notifica e comunicare lo stato con i client. Pertanto, il sistema di controllo dello stato dovrebbe ogni volta inviare lo snapshot completo dello stato corrente. 100.000 stati di integrità del server, ciascuno rappresentato da un bit, sarebbero solo un payload di 12,5 KB. Indipendentemente dal fatto che non ci siano server guasti, o che lo siano tutti, il sistema di controllo dello stato esegue un lavoro costante e le modifiche rapide e di grandi dimensioni non rappresentano una minaccia per la stabilità del sistema. Questo è in realtà il modo in cui Amazon Route 53 gestisce i controlli dell'integrità degli endpoint (come gli indirizzi IP) per stabilire come gli utenti finali vengono instradati verso di loro.

Livello di rischio associato se questa best practice non fosse adottata: Bassa

Guida all'implementazione

- Esegui un lavoro costante in modo che i sistemi non falliscano quando si verificano cambiamenti rapidi e significativi nel carico.
- Implementazione di dipendenze "loosely coupled". Le dipendenze come sistemi di accodamento, sistemi di streaming, flussi di lavoro e sistemi di bilanciamento del carico sono "loosely

coupled" (con accoppiamento debole). L'accoppiamento debole aiuta a isolare il comportamento di un componente dagli altri componenti che dipendono da esso, aumentando la resilienza e l'agilità.

- The Amazon Builders' Library: Reliability, constant work, and a good cup of coffee
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small (Chiudere i cicli e aprire le menti: come prendere il controllo dei sistemi, grandi e piccoli), include lavoro costante (ARC337)
  - Per l'esempio di un sistema di controllo dell'integrità che monitora 100.000 server, progetta i carichi di lavoro in modo che le dimensioni dei payload rimangano costanti indipendentemente dal numero di successi o di fallimenti.

Risorse

Documenti correlati:

- Amazon EC2: garantire l'idempotenza
- The Amazon Builders' Library: Difficoltà dei sistemi distribuiti
- The Amazon Builders' Library: Reliability, constant work, and a good cup of coffee

Video correlati:

- AWS New York Summit 2019: Intro to Event-driven Architectures and Amazon EventBridge (Introduzione alle architetture guidate dagli eventi e ad Amazon EventBridge) (MAD205)
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small (Chiudere i cicli e aprire le menti: come prendere il controllo dei sistemi, grandi e piccoli), include lavoro costante (ARC337)
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small (Chiudere i cicli e aprire le menti: come prendere il controllo dei sistemi, grandi e piccoli), sono inclusi accoppiamento debole, lavoro costante e stabilità statica (ARC337)
- AWS re:Invent 2019: passare alle architetture basate sugli eventi (SVS308)

REL04-BP04 Rendere tutte le risposte idempotenti

Un servizio idempotente promette il completamento di ogni richiesta esattamente una volta, in modo tale che effettuare più richieste identiche abbia lo stesso effetto di effettuare una singola richiesta. Un servizio idempotente semplifica ad un client l'implementazione di nuovi tentativi senza temere

che una richiesta venga elaborata erroneamente più volte. Per eseguire questa operazione, i client possono inviare richieste API con un token di idempotenza: viene utilizzato lo stesso token ogni volta che si ripete la richiesta. Un'API del servizio idempotente utilizza il token per restituire una risposta identica a quella restituita la prima volta che la richiesta è stata completata.

In un sistema distribuito, è facile eseguire un'operazione al massimo una volta (il client effettua una sola richiesta) o almeno una volta (la richiesta continua finché il client non ottiene la conferma dell'esito positivo). Tuttavia, è difficile garantire che un'operazione sia idempotente, il che significa che viene eseguita esattamente una volta, in modo tale che effettuare più richieste identiche abbia lo stesso effetto di effettuare una singola richiesta. Utilizzando i token di idempotenza nelle API, i servizi possono ricevere una richiesta di mutazione una o più volte senza creare record duplicati o effetti collaterali.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

- Rendi tutte le risposte idempotenti. Un servizio idempotente promette il completamento di ogni richiesta esattamente una volta, in modo tale che effettuare più richieste identiche abbia lo stesso effetto di effettuare una singola richiesta.
  - I client possono inviare richieste API con un token di idempotenza: viene utilizzato lo stesso token ogni volta che si ripete la richiesta. Un'API del servizio idempotente utilizza il token per restituire una risposta identica a quella restituita la prima volta che la richiesta è stata completata.
    - Amazon EC2: Ensuring Idempotency (EC2: garantire l'idempotenza)

Risorse

Documenti correlati:

- Amazon EC2: Ensuring Idempotency (EC2: garantire l'idempotenza)
- The Amazon Builders' Library: Difficoltà dei sistemi distribuiti
- The Amazon Builders' Library: Reliability, constant work, and a good cup of coffee

Video correlati:

- AWS New York Summit 2019: Intro to Event-driven Architectures and Amazon EventBridge (AWS New York Summit 2019: Introduzione alle architetture guidate dagli eventi e ad Amazon EventBridge) (MAD205)

- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small (Chiudere i cicli e aprire le menti: come prendere il controllo dei sistemi, grandi e piccoli) (sono inclusi accoppiamento debole, lavoro costante e stabilità statica) (ARC337)
- AWS re:Invent 2019: Moving to event-driven architectures (Passare alle architetture basate sugli eventi) (SVS308)

REL 5. Come si progettano le interazioni in un sistema distribuito per mitigare o affrontare gli errori?

I sistemi distribuiti si basano sulle reti di comunicazione per interconnettere i componenti (ad esempio server o servizi). Il carico di lavoro deve funzionare in modo affidabile nonostante la perdita o la latenza dei dati su queste reti. I componenti del sistema distribuito devono funzionare in modo da non influire negativamente su altri componenti o sul carico di lavoro. Queste best practice permettono ai carichi di lavoro di tollerare le sollecitazioni o i guasti, recuperare più rapidamente e mitigare l'impatto di tali problemi. Il risultato è un miglioramento del tempo medio di ripristino (MTTR).

Best practice

- REL05-BP01 Implementazione della normale riduzione delle prestazioni per trasformare le dipendenze forti applicabili in dipendenze deboli
- REL05-BP02 Richieste di limitazione (della larghezza di banda della rete)
- REL05-BP03 Controllo e limitazione delle chiamate di ripetizione
- REL05-BP04 Anticipazione degli errori e limitazione delle code
- REL05-BP05 Impostazione dei timeout dei client
- REL05-BP06 Utilizzo dei sistemi stateless laddove possibile
- REL05-BP07 Implementazione di leve di emergenza

REL05-BP01 Implementazione della normale riduzione delle prestazioni per trasformare le dipendenze forti applicabili in dipendenze deboli

I componenti dell'applicazione devono continuare a svolgere la loro funzione principale anche se le dipendenze non sono disponibili. Potrebbero fornire dati leggermente obsoleti, dati alternativi o addirittura nessun dato. Ciò garantisce che la funzionalità complessiva del sistema sia ostacolata solo in minima parte da errori localizzati, garantendo al contempo il valore aziendale intrinseco.

Risultato desiderato: quando le dipendenze di un componente non sono integre, il componente stesso può comunque funzionare, anche se non in modo ottimale. Le modalità di errore dei

componenti devono essere considerate come funzionamenti normali. I flussi di lavoro devono essere progettati in modo tale che questi errori non conducano a un fallimento completo o almeno a stati prevedibili e recuperabili.

Anti-pattern comuni:

- Mancata identificazione della funzionalità aziendale di base necessaria. Mancata verifica del funzionamento dei componenti anche in caso di errori di dipendenza.
- Mancata restituzione di dati sugli errori o quando solo una delle dipendenze non è disponibile e possono comunque essere restituiti risultati parziali.
- Creazione di uno stato incoerente quando una transazione fallisce parzialmente.
- Mancata disponibilità di alternative per accedere a un archivio di parametri centralizzato.
- Invalidare o svuotare lo stato locale a seguito di un aggiornamento non riuscito senza considerare le conseguenze di tale operazione.

Vantaggi dell'adozione di questa best practice: la normale riduzione delle prestazioni migliora la disponibilità del sistema nel suo complesso e conserva la funzionalità delle funzioni più importanti anche in caso di errori.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

L'implementazione di una normale riduzione delle prestazioni aiuta a ridurre al minimo l'impatto degli errori di dipendenza sul funzionamento dei componenti. Idealmente, un componente rileva gli errori nelle dipendenze e trova soluzioni alternative in modo da avere un impatto minimo sugli altri componenti o clienti.

Progettare per una normale riduzione delle prestazioni significa considerare le potenziali modalità di errore durante la progettazione delle dipendenze. Per ogni modalità di errore, disponi di un modo per fornire la maggior parte delle funzionalità o almeno quelle più critiche del componente a chiamanti o clienti. Queste considerazioni possono diventare requisiti aggiuntivi testabili e verificabili. Idealmente, un componente è in grado di svolgere la sua funzione principale in modo accettabile anche in caso di errore di una o più dipendenze.

Questa è una discussione di carattere tanto commerciale quanto tecnico. Tutti i requisiti aziendali sono importanti e devono essere soddisfatti, se possibile. Tuttavia, ha ancora senso chiedersi cosa dovrebbe succedere quando non tutti i requisiti possono essere soddisfatti. Un sistema può essere

progettato per essere disponibile e coerente, ma nelle circostanze in cui è necessario eliminare un requisito, qual è quello più importante? Per l'elaborazione dei pagamenti, potrebbe essere la coerenza. Per un'applicazione in tempo reale, potrebbe essere la disponibilità. Per un sito Web rivolto ai clienti, la risposta può dipendere dalle aspettative dei clienti.

Il significato di ciò dipende dai requisiti del componente e da ciò che dovrebbe essere considerato la sua funzione principale. Ad esempio:

- un sito di e-commerce potrebbe visualizzare dati provenienti da più sistemi diversi, come consigli personalizzati, prodotti con il punteggio più alto e lo stato degli ordini dei clienti sulla pagina di destinazione. Quando in un sistema upstream si verifica un errore, ha comunque senso mostrare tutto il resto, invece di mostrare una pagina di errore a un cliente.
- Un componente che esegue la scrittura in batch può continuare a elaborare un batch se una delle singole operazioni fallisce. Dovrebbe essere semplice implementare un meccanismo di ripetizione dei tentativi. A tale scopo, è sufficiente restituire al chiamante informazioni su quali operazioni hanno avuto successo, quali e perché non sono riuscite, oppure inserendo le richieste non riuscite in una coda DLQ per implementare nuovi tentativi asincroni. Anche le informazioni sulle operazioni non riuscite devono essere registrate.
- Un sistema che elabora le transazioni deve verificare che vengano eseguiti tutti gli aggiornamenti o nessun aggiornamento. Per le transazioni distribuite, il modello Saga può essere utilizzato per ripristinare le operazioni precedenti nel caso in cui fallisca un'operazione successiva della stessa transazione. Qui, la funzione principale è mantenere la coerenza.
- I sistemi critici dal punto di vista temporale dovrebbero essere in grado di gestire le dipendenze che non rispondono in modo tempestivo. In questi casi, è possibile utilizzare lo schema dell'interruttore. Quando inizia a verificarsi il timeout delle risposte di una dipendenza, il sistema può passare a uno stato chiuso in cui non vengono effettuate chiamate aggiuntive.
- Un'applicazione può leggere i parametri da un archivio di parametri. Può essere utile creare immagini di container con un set di parametri predefinito e utilizzarli nel caso in cui l'archivio dei parametri non sia disponibile.

Si noti che i percorsi seguiti in caso di errore dei componenti devono essere testati e devono essere significativamente più semplici del percorso primario. In genere, è consigliabile evitare il fallback.

Passaggi dell'implementazione

Identifica le dipendenze esterne e interne. Considera i tipi di errore che si possono verificare nelle dipendenze. Considera i modi per ridurre al minimo l'impatto negativo sui sistemi upstream e downstream e sui clienti durante questi errori.

Di seguito è riportato un elenco di dipendenze e di come ridurre normalmente le prestazioni quando si verifica un errore a livello di dipendenze:

1. Errore parziale delle dipendenze: un componente può effettuare più richieste ai sistemi downstream, sia come richieste multiple a un sistema sia come richiesta a più sistemi. A seconda del contesto aziendale, possono essere appropriate diverse modalità di gestione (per maggiori dettagli, consulta gli esempi precedenti nella Guida all'implementazione).
2. Un sistema downstream non è in grado di elaborare le richieste a causa del carico elevato: se le richieste a un sistema downstream hanno costantemente esito negativo, non ha senso continuare a riprovare. Ciò può creare un carico aggiuntivo su un sistema già sovraccarico e rendere più difficile il ripristino. Qui è possibile utilizzare lo schema dell'interruttore, che monitora le chiamate non riuscite a un sistema downstream. Se un numero elevato di chiamate ha esito negativo, interromperà l'invio di altre richieste al sistema downstream e solo occasionalmente lascerà passare le chiamate per verificare se il sistema downstream è nuovamente disponibile.
3. Un archivio di parametri non è disponibile: per trasformare un archivio di parametri, è possibile utilizzare la cache delle dipendenze a protezione debole o i valori predefiniti integri inclusi nelle immagini del container o del computer. Tieni presente che queste impostazioni predefinite devono essere costantemente aggiornate e incluse nelle suite di test.
4. Un servizio di monitoraggio o altra dipendenza non funzionale non è disponibile: se un componente non è in grado di inviare a intermittenza log, metriche o tracce a un servizio di monitoraggio centralizzato, spesso è meglio continuare a eseguire le funzioni aziendali come al solito. Non registrare o eseguire il push delle metriche in modo invisibile all'utente per un lungo periodo di tempo spesso non è una procedura accettabile. Inoltre, in alcuni casi d'uso potrebbero essere necessari dati di controllo completi per soddisfare i requisiti di conformità.
5. Un'istanza primaria di un database relazionale potrebbe non essere disponibile: Amazon Relational Database Service, come quasi tutti i database relazionali, può avere solo un'istanza di scrittura primaria. Questo crea un unico punto di errore per i carichi di lavoro di scrittura e rende più difficile il dimensionamento. Questo problema può essere parzialmente mitigato utilizzando una configurazione Multi-AZ per una disponibilità elevata o Amazon Aurora Serverless per un migliore dimensionamento. Per requisiti di disponibilità molto elevati, può essere logico non fare affatto affidamento sull'istanza di scrittura primaria. Per le query che si limitano a leggere, è

possibile utilizzare repliche di lettura, che forniscono ridondanza e la possibilità di dimensionare non solo verticalmente, ma anche orizzontalmente. Le operazioni di scrittura possono essere memorizzate nel buffer, ad esempio in una coda Amazon Simple Queue Service, in modo che le richieste di scrittura dei clienti possano comunque essere accettate anche se l'istanza primaria non è temporaneamente disponibile.

Risorse

Documenti correlati:

- Amazon API Gateway: throttling delle richieste API per migliorare la velocità di trasmissione effettiva
- CircuitBreaker (riepilogo dal libro Circuit Breaker da "Release It!")
- Ripetizione dei tentativi in caso di errore e backoff esponenziale in AWS
- Michael Nygard "Release It! Design and Deploy Production-Ready Software"
- The Amazon Builders' Library: Evitare il fallback nei sistemi distribuiti
- The Amazon Builders' Library: Evitare insormontabili backlog di code
- The Amazon Builders' Library: Sfide e strategie del caching
- The Amazon Builders' Library: Timeout, nuovi tentativi e backoff con jitter

Video correlati:

- Retry, backoff, and jitter: AWS re:Invent 2019: Introducing The Amazon Builders' Library (Presentazione della libreria dei costruttori di Amazon) (DOP328)

Esempi correlati:

- Corso Well-Architected: Level 300: Implementing Health Checks and Managing Dependencies to Improve Reliability

REL05-BP02 Richieste di limitazione (della larghezza di banda della rete)

Limita le richieste per mitigare l'esaurimento delle risorse dovuto ad aumenti imprevisti della domanda. Le richieste inferiori alla percentuale di limitazione (della larghezza di banda della rete) vengono elaborate mentre quelle che superano il limite definito vengono rifiutate con un messaggio che indica che la richiesta è stata limitata.

Risultato desiderato: i picchi di volume di grandi dimensioni dovuti a improvvisi aumenti del traffico dei clienti, attacchi di flooding o tempeste di ripetizioni dei tentativi sono mitigati dalla limitazione (della larghezza di banda della rete) delle richieste, che consente ai carichi di lavoro di continuare la normale elaborazione del volume di richieste supportato.

Anti-pattern comuni:

- Le accelerazioni degli endpoint API non sono implementate o vengono implementate in base ai valori predefiniti senza considerare i volumi previsti.
- Gli endpoint delle API non sono sottoposti a test di carico né i limiti relativi alla limitazione (della larghezza di banda della rete) vengono testati.
- Limitazione delle tariffe delle richieste senza considerare le dimensioni o la complessità delle richieste.
- Verifica delle percentuali massime di richieste o delle dimensioni massime delle richieste, senza però testarle congiuntamente.
- Le risorse non vengono fornite entro gli stessi limiti stabiliti durante i test.
- I piani di utilizzo non sono stati configurati o considerati per gli utenti di API Application to Application (A2A).
- Gli utenti di code con dimensionamento orizzontale non hanno configurato le impostazioni di simultaneità massima.
- La limitazione della velocità per indirizzo IP non è stata implementata.

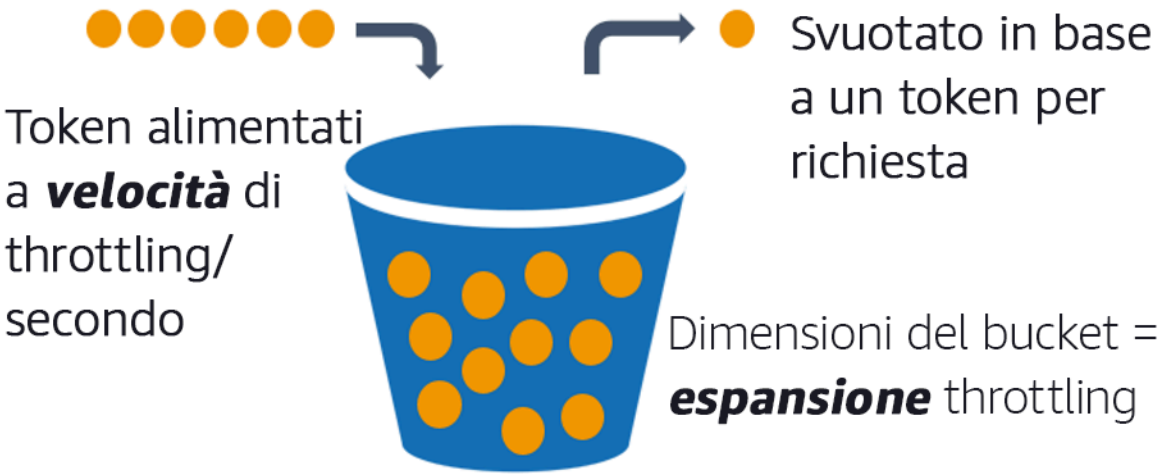
Vantaggi dell'adozione di questa best practice: i carichi di lavoro che stabiliscono limiti di accelerazione sono in grado di funzionare normalmente ed elaborare correttamente il caricamento delle richieste accettate in presenza di picchi di volume imprevisti. I picchi improvvisi o prolungati di richieste alle API e alle code vengono limitati e non esauriscono le risorse di elaborazione delle richieste. I limiti tariffari vincolano i richiedenti in modo che elevati volumi di traffico provenienti da un utente di un indirizzo IP o API specifico non esauriscano le risorse e influiscano sugli altri utenti.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

I servizi devono essere progettati per elaborare una capacità nota di richieste; tale capacità può essere stabilita mediante test di carico. Se le percentuali di arrivo delle richieste superano i limiti, la risposta appropriata segnala che una richiesta è stata limitata. Ciò consente all'utente di gestire l'errore e riprovare in un secondo momento.

Quando il servizio richiede un'implementazione della limitazione (della larghezza di banda della rete), prendi in considerazione l'implementazione dell'algoritmo token bucket, in cui un token conta come una richiesta. I token vengono alimentati a una specifica velocità di throttling al secondo e svuotati in modo asincrono in base a un token per richiesta.



Algoritmo token bucket.

Amazon API Gateway implementa l'algoritmo token bucket in base ai limiti dell'account e della regione e può essere configurato per cliente con piani di utilizzo. Inoltre, Amazon Simple Queue Service (Amazon SQS) e Amazon Kinesis possono memorizzare le richieste nel buffer per livellare la frequenza delle richieste e consentire percentuali di limitazione più elevati per le richieste che possono essere soddisfatte. Infine, puoi implementare la limitazione della velocità con AWS WAF per limitare utenti di API specifici che generano carichi insolitamente elevati.

Passaggi dell'implementazione

Puoi configurare API Gateway con limiti di limitazione (della larghezza di banda della rete) per le tue API e restituire errori `429 - Troppe richieste` in caso di superamento dei limiti. Puoi utilizzare AWS WAF con gli endpoint API Gateway e AWS AppSync per abilitare la limitazione della velocità per indirizzo IP. Inoltre, laddove il sistema può tollerare l'elaborazione asincrona, è possibile inserire i messaggi in una coda o in un flusso per velocizzare le risposte ai client del servizio, il che consente di aumentare le velocità.

Con l'elaborazione asincrona, una volta configurato Amazon SQS come origine degli eventi per AWS Lambda, è possibile configurare la simultaneità massima per evitare che percentuali elevate di eventi consumino la quota di esecuzione simultanea disponibile dell'account necessaria per altri servizi nel carico di lavoro o nell'account.

Sebbene API Gateway fornisca un'implementazione gestita dell'algoritmo token bucket, nei casi in cui non sia possibile utilizzare API Gateway, puoi sfruttare le implementazioni open source specifiche del linguaggio (consulta gli esempi correlati nella sezione Risorse) dell'algoritmo token bucket per i tuoi servizi.

- Analizza e configura i valori di limitazione (della larghezza di banda della rete) API Gateway a livello di account per regione, API per fase e chiave API per livelli del piano di utilizzo.
- Applica le regole di limitazione (della larghezza di banda della rete) AWS WAF sugli endpoint API Gateway e AWS AppSync come prevenzione degli attacchi flood e per bloccare gli IP pericolosi. Le regole di limitazione (della larghezza di banda della rete) possono anche essere configurate su chiavi API AWS AppSync per gli utenti A2A.
- Valuta se hai bisogno di più controllo sulla limitazione della larghezza di banda della rete rispetto al controllo sulla limitazione della velocità per le API AWS AppSync e, in tal caso, configura un API Gateway davanti all'endpoint AWS AppSync.
- Quando le code Amazon SQS sono impostate come trigger per gli utenti della coda Lambda, imposta la simultaneità massima su un valore che elabora in misura sufficiente a soddisfare gli obiettivi dei livelli di servizio ma non consuma i limiti di simultaneità che influiscono su altre funzioni Lambda. Valuta la possibilità di impostare la simultaneità riservata su altre funzioni Lambda nello stesso account e nella stessa regione quando utilizzi le code con Lambda.
- Utilizza API Gateway con integrazioni di servizi native per Amazon SQS o Kinesis per memorizzare le richieste nel buffer.
- Se non puoi utilizzare API Gateway, consulta le librerie specifiche della lingua per implementare l'algoritmo token bucket per il tuo carico di lavoro. Controlla la sezione degli esempi e cerca una libreria adatta.
- Verifica i limiti che intendi impostare o che prevedi di incrementare e documenta i limiti testati.
- Non aumentare i limiti oltre i valori stabiliti durante i test. Quando si aumenta un limite, verifica che le risorse assegnate siano equivalenti o superiori a quelle degli scenari di test prima di applicare l'aumento.

Risorse

Best practice correlate:

- REL04-BP03 Esecuzione di un lavoro costante
- REL05-BP03 Controllo e limitazione delle chiamate di ripetizione

Documenti correlati:

- Amazon API Gateway: throttling delle richieste API per migliorare la velocità di trasmissione effettiva
- AWS WAF: istruzione delle regole basate sulla frequenza
- Introduzione alla simultaneità massima di AWS Lambda in caso di utilizzo Amazon SQS come origine degli eventi
- AWS Lambda: simultaneità massima

Esempi correlati:

- Le tre regole AWS WAF più importanti basate sulla velocità
- Java Bucket4j
- Algoritmo token bucket per Python
- Algoritmo token bucket a livello di nodo
- Limitazione della velocità di threading del sistema .NET

Video correlati:

- Implementazione delle best practice di sicurezza dell'API GraphQL con AWS AppSync

Strumenti correlati:

- Amazon API Gateway
- AWS AppSync
- Amazon SQS
- Amazon Kinesis
- AWS WAF

REL05-BP03 Controllo e limitazione delle chiamate di ripetizione

Utilizza il backoff esponenziale per rieseguire le richieste a intervalli progressivamente più lunghi tra i singoli nuovi tentativi. Introduci il jitter tra i tentativi per randomizzare gli intervalli di ripetizione. Limita il numero massimo di tentativi.

Risultato desiderato: I componenti tipici di un sistema software distribuito includono server, sistemi di bilanciamento del carico, database e server DNS. Durante il normale funzionamento, questi componenti possono rispondere alle richieste con errori temporanei o limitati e anche errori che sarebbero persistenti indipendentemente dai nuovi tentativi. Quando i client effettuano richieste ai servizi, le richieste consumano risorse tra cui memoria, thread, connessioni, porte o altre risorse limitate. Controllare e limitare i nuovi tentativi è una strategia per liberare risorse e ridurne al minimo il consumo in modo che i componenti del sistema sottoposti a stress non vengano sovraccaricati.

Quando vanno in timeout o ricevono risposte di errore, le richieste client devono decidere se eseguire o meno nuovi tentativi. Se vengono eseguiti nuovi tentativi, questi verranno eseguiti con un backoff esponenziale con jitter e un numero massimo di tentativi. Di conseguenza, i servizi e i processi back-end riducono il carico e i tempi di riparazione automatica, con un conseguente ripristino più rapido e una corretta gestione delle richieste.

Anti-pattern comuni:

- Implementazione di nuovi tentativi senza aggiungere il backoff esponenziale, il jitter e il numero massimo di tentativi. Il backoff e il jitter aiutano a evitare picchi di traffico artificiali dovuti a tentativi involontariamente coordinati a intervalli standard.
- Implementazione di nuovi tentativi senza testarne gli effetti o assunzione che i nuovi tentativi siano già integrati in un SDK senza testare gli scenari di ripetizione dei tentativi.
- La mancata comprensione dei codici di errore pubblicati nelle dipendenze porta a ritentare tutti gli errori, compresi quelli la cui causa è chiara e indica la mancanza di autorizzazione, un errore di configurazione o un'altra condizione che prevedibilmente non si risolverà senza un intervento manuale.
- Mancata gestione delle best practice di osservabilità, compresi il monitoraggio e la segnalazione di ripetuti guasti del servizio, in modo che i problemi sottostanti siano resi noti e possano essere risolti.
- Sviluppo di meccanismi di ripetizione personalizzati quando le funzionalità di ripetizione dei tentativi integrate o di terze parti sono sufficienti.
- Riprovare su più livelli dello stack di applicazioni in modo da accrescere in modo significativo i nuovi tentativi e pertanto da consumare ulteriormente le risorse in una tempesta di ripetizioni dei tentativi. Assicurati di comprendere in che modo questi errori influiscono sulla tua applicazione e sulle dipendenze su cui fai affidamento, quindi implementa i nuovi tentativi a un solo livello.
- Riesecuzione delle chiamate dei servizi non idempotenti, con effetti collaterali imprevisti come risultati duplicati.

Vantaggi dell'adozione di questa best practice: i nuovi tentativi aiutano i client a ottenere i risultati desiderati quando le richieste non riescono, ma consumano più tempo del server per ottenere le risposte corrette desiderate. Quando gli errori sono rari o transitori, i nuovi tentativi funzionano correttamente. Quando gli errori sono causati da un sovraccarico di risorse, i nuovi tentativi possono peggiorare le cose. L'aggiunta di un backoff esponenziale con jitter ai tentativi dei client consente ai server di recuperare risorse quando gli errori sono causati dal sovraccarico delle risorse. Il jitter evita l'allineamento delle richieste in picchi e il backoff riduce l'aumento del carico causato dall'aggiunta di nuovi tentativi al normale carico delle richieste. Infine, è importante configurare un numero massimo di tentativi o il tempo trascorso per evitare la creazione di backlog che producono errori metastabili.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Controlla e limita le chiamate riproposte. Utilizza il backoff esponenziale per eseguire nuovi tentativi dopo intervalli progressivamente più lunghi. Introduci il jitter per randomizzare gli intervalli di ripetizione e limitare il numero massimo di tentativi.

Alcuni AWS SDK implementano i nuovi tentativi e il backoff esponenziale per impostazione predefinita. Usa queste implementazioni AWS integrate laddove applicabile nel tuo carico di lavoro. Implementa una logica simile nel tuo carico di lavoro quando chiami servizi idempotenti e i cui tentativi migliorano la disponibilità dei client. Potrai decidere quali sono i timeout e quando cessare i tentativi in base al tuo caso d'uso. Crea ed esegui scenari di test per quei casi d'uso relativi ai nuovi tentativi.

Passaggi dell'implementazione

- Determina il livello ottimale nello stack di applicazioni per implementare nuovi tentativi per i servizi su cui si basa l'applicazione.
- Presta attenzione agli SDK esistenti che implementano strategie collaudate di ripetizione dei tentativi con backoff esponenziale e jitter per la lingua prescelta, e preferisci queste soluzioni anziché scrivere implementazioni personalizzate.
- Verifica che i servizi siano idempotenti prima di implementare nuovi tentativi. Una volta implementati i nuovi tentativi, assicurati che siano testati e che vengano regolarmente eseguiti in produzione.
- Quando chiami le API del servizio AWS, utilizza gli AWS SDK e AWS CLI e analizza le opzioni di configurazione dei nuovi tentativi. Determina se le impostazioni predefinite sono adatte al tuo caso d'uso, esegui i test e regola i valori secondo necessità.

Risorse

Best practice correlate:

- REL04-BP04 Rendere tutte le risposte idempotenti
- REL05-BP02 Richieste di limitazione (della larghezza di banda della rete)
- REL05-BP04 Anticipazione degli errori e limitazione delle code
- REL05-BP05 Impostazione dei timeout dei client
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Documenti correlati:

- Ripetizione dei tentativi in caso di errore e backoff esponenziale in AWS
- The Amazon Builders' Library: Timeout, nuovi tentativi e backoff con jitter
- Exponential Backoff and Jitter (Jitter e backoff esponenziale)
- Rendere sicuri i tentativi con API idempotenti

Esempi correlati:

- Spring Retry
- Resilience4j Retry

Video correlati:

- Retry, backoff, and jitter: AWS re:Invent 2019: Introducing The Amazon Builders' Library (Presentazione della libreria dei costruttori di Amazon) (DOP328)

Strumenti correlati:

- AWS SDKs and Tools: Retry behavior (AWS SDK e strumenti: comportamento dei tentativi)
- AWS Command Line Interface: tentativi AWS CLI

REL05-BP04 Anticipazione degli errori e limitazione delle code

Se un servizio non è in grado di rispondere correttamente a una richiesta, anticipa l'errore (fail fast). Ciò consente il rilascio delle risorse associate a una richiesta e permette al servizio di recuperare

le risorse se queste sono in esaurimento. L'anticipazione degli errori (fail fast) è un modello di progettazione software consolidato che può essere usato per creare carichi di lavoro altamente affidabili nel cloud. Anche l'accodamento è un modello di integrazione aziendale consolidato che può semplificare il carico e consentire ai client di rilasciare risorse quando l'elaborazione asincrona può essere tollerata. Quando un servizio è in grado di rispondere correttamente in condizioni normali ma fallisce quando la frequenza delle richieste è troppo alta, utilizza una coda per memorizzare le richieste nel buffer. Tuttavia, non consentire la creazione di backlog di code lunghe che possono comportare l'elaborazione di richieste obsolete già dismesse dal client.

Risultato desiderato: Quando i sistemi rilevano conflitti a livello di risorse, timeout, eccezioni o errori che rendono irraggiungibili gli obiettivi dei livelli di servizio, le strategie di anticipazione degli errori (fail fast) consentono un ripristino più rapido del sistema. I sistemi che devono assorbire i picchi di traffico e sono in grado di gestire l'elaborazione asincrona possono migliorare l'affidabilità consentendo ai client di rilasciare rapidamente le richieste utilizzando le code per archiviare le richieste nei servizi di back-end. Quando le richieste vengono memorizzate nei buffer delle code, vengono implementate strategie di gestione delle code per evitare backlog ingestibili.

Anti-pattern comuni:

- Implementazione delle code di messaggi ma non la configurazione delle code DLQ o degli allarmi nei volumi DLQ per rilevare quando un sistema è in errore.
- Mancata misurazione dell'età dei messaggi in una coda, misurazione della latenza per capire quando gli utenti della coda sono in ritardo o generano errori che causano un nuovo tentativo.
- Mancata cancellazione dei messaggi nel backlog da una coda quando non è più necessario elaborare questi messaggi se l'azienda non lo richiede più.
- La configurazione delle code First in First Out (FIFO) quando le code Last In First Out (LIFO) soddisferebbe meglio le esigenze dei client, ad esempio quando non sono richiesti ordini rigorosi e l'elaborazione dei backlog sta ritardando tutte le richieste nuove e urgenti, con conseguente violazione dei livelli di servizio per tutti i client.
- Esposizione delle code interne ai client, invece dell'esposizione delle API che gestiscono l'acquisizione del lavoro e l'inserimento delle richieste in code interne.
- Combinazione di un numero eccessivo di tipi di richieste di lavoro in un'unica coda; ciò può aggravare le condizioni dei backlog in seguito alla distribuzione delle richieste di risorse tra i tipi di richiesta.
- Elaborazione di richieste complesse e semplici nella stessa coda, nonostante siano necessari monitoraggio, timeout e allocazioni di risorse diversi.

- Mancata convalida degli input o utilizzo di asserzioni per implementare meccanismi di anticipazione degli errori (fail fast) nel software che generano eccezioni a componenti di livello superiore in grado di gestire normalmente gli errori.
- Mancata rimozione delle risorse in errore dall'instradamento delle richieste, soprattutto quando gli errori generano risultati sia positivi che negativi dovuti ad arresti anomali e riavvii, errori intermittenti a livello di dipendenze, capacità ridotta o perdita di pacchetti di rete.

Vantaggi dell'adozione di questa best practice: I sistemi con anticipazione degli errori sono più facili da sottoporre al debug e alla correzione degli errori e spesso presentano problemi di codifica e configurazione prima che le versioni vengano pubblicate in produzione. I sistemi che incorporano strategie di accodamento efficaci forniscono maggiore resilienza e affidabilità in caso di picchi di traffico e di condizioni intermittenti di errore del sistema.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Le strategie di anticipazione degli errori possono essere codificate in soluzioni software e configurate nell'infrastruttura. Oltre all'anticipazione degli errori (fail fast), le code sono una tecnica semplice ma affidabile di definizione dell'architettura che consente il caricamento senza problemi di componenti disaccoppiati del sistema. Amazon CloudWatch fornisce funzionalità per il monitoraggio e la segnalazione di guasti. Una volta accertato il malfunzionamento di un sistema, è possibile ricorrere a strategie di mitigazione, ad esempio per evitare problemi dovuti a risorse danneggiate. Quando i sistemi implementano le code con Amazon SQS e altre tecnologie di accodamento, per semplificare il caricamento, devono valutare come gestire i backlog e gli errori di utilizzo dei messaggi.

Passaggi dell'implementazione

- Implementa asserzioni programmatiche o metriche specifiche nel tuo software e utilizzale per avvisare esplicitamente in caso di problemi a livello di sistema. Amazon CloudWatch ti aiuta a creare metriche e allarmi in base al modello di log delle applicazioni e alla strumentazione SDK.
- Usa le metriche CloudWatch e gli allarmi per eseguire il failover per le risorse danneggiate responsabili dell'incremento della latenza dell'elaborazione o che ripetutamente non riescono a elaborare le richieste.
- Utilizza l'elaborazione asincrona. A tale scopo, progetta API in grado di accettare le richieste e aggiungere richieste alle code interne mediante Amazon SQS e, quindi, rispondere al client che genera il messaggio con un messaggio di successo, in modo che il client possa rilasciare risorse e passare ad altre attività mentre gli utenti nella coda di back-end elaborano le richieste.

- Misura e monitora la latenza di elaborazione delle code generando una metrica CloudWatch ogni volta che escludi un messaggio da una coda confrontandolo con il timestamp del messaggio.
- Quando gli errori impediscono la corretta elaborazione dei messaggi o il traffico aumenta a livelli tali da impedirne l'elaborazione in base agli accordi sul livello di servizio, escludi il traffico obsoleto o in eccesso indirizzandolo a una coda per il traffico eccedente. Ciò consente l'elaborazione prioritaria del nuovo processo e del processo più vecchio quando si rende disponibile nuova capacità. Questa tecnica è un'approssimazione dell'elaborazione LIFO e consente la normale elaborazione del sistema per tutti i nuovi processi.
- Usa le code DLQ o le code di reindirizzamento per spostare i messaggi che non possono essere elaborati dal backlog in una posizione che può essere ricercata e risolta in un secondo momento.
- Riprova o, se possibile, elimina i vecchi messaggi confrontandoli con il timestamp del messaggio ed eliminando i messaggi che non sono più rilevanti per il client richiedente.

Risorse

Best practice correlate:

- REL04-BP02 Implementazione di dipendenze "loosely coupled"
- REL05-BP02 Richieste di limitazione (della larghezza di banda della rete)
- REL05-BP03 Controllo e limitazione delle chiamate di ripetizione
- REL06-BP02 Definizione e calcolo dei parametri (aggregazione)
- REL06-BP07 Monitoraggio del tracciamento end-to-end delle richieste attraverso il sistema

Documenti correlati:

- Evitare backlog di coda insormontabili
- Anticipazione degli errori (fail fast)
- Come posso prevenire un aumento del backlog dei messaggi nella mia coda Amazon SQS?
- Elastic Load Balancing: spostamento zonale
- Sistema di controllo Amazon Route 53 per il ripristino di applicazioni: controllo dell'instradamento per il failover del traffico

Esempi correlati:

- Modelli di integrazione aziendale: canale DLQ

Video correlati:

- AWS re:Invent 2022 - Operating highly available Multi-AZ applications (Esecuzione di applicazioni multi-AZ a disponibilità elevata)

Strumenti correlati:

- Amazon SQS
- Amazon MQ
- AWS IoT Core
- Amazon CloudWatch

REL05-BP05 Impostazione dei timeout dei client

Imposta i timeout in modo appropriato per connessioni e richieste, verificali sistematicamente e non fare affidamento sui valori predefiniti perché non fanno riferimento alle specifiche del carico di lavoro.

Risultato desiderato: I timeout dei client devono considerare il costo per client, server e carico di lavoro associato all'attesa di richieste il cui completamento richiede una quantità di tempo anomala. Poiché non è possibile conoscere la causa esatta di un timeout, i client devono fare riferimento ai servizi per sviluppare ipotesi sulle cause probabili e sui timeout appropriati.

Il timeout delle connessioni client si verifica in base ai valori configurati. Dopo aver rilevato un timeout, i client decidono di riprovare o aprire un interruttore. Questi modelli evitano la generazione di richieste che potrebbero aggravare una condizione di errore sottostante.

Anti-pattern comuni:

- Non essere a conoscenza dei timeout di sistema o dei timeout predefiniti.
- Non essere a conoscenza dei normali tempi di completamento delle richieste.
- Non essere a conoscenza delle possibili cause dei tempi anomali necessari per il completamento delle richieste o dei costi in termini di prestazioni di client, servizio o carico di lavoro associati all'attesa di tali completamenti.
- Non essere consapevoli della probabilità che una rete danneggiata causi un errore di esecuzione della richiesta solo al raggiungimento del timeout, nonché dei costi in termini di prestazioni del client e del carico di lavoro derivanti dalla mancata adozione di un timeout più breve.
- Non testare gli scenari di timeout sia per le connessioni che per le richieste.

- Impostazione di timeout troppo elevati, che può comportare lunghi tempi di attesa e aumentare l'utilizzo delle risorse.
- Impostazione di timeout troppo bassi, con conseguenti errori artificiali.
- Mancata verifica degli schemi per gestire gli errori di timeout per chiamate remote come interruttori e nuovi tentativi.
- Non considerare il monitoraggio delle percentuali di errore delle chiamate dei servizi, degli obiettivi del livello di servizio per la latenza e dei valori anomali della latenza. Queste metriche possono fornire informazioni sui timeout restrittivi o permissivi.

Vantaggi dell'adozione di questa best practice: I timeout delle chiamate remote sono configurati e i sistemi sono progettati per gestirli correttamente, in modo da preservare le risorse quando le chiamate remote rispondono in modo eccessivamente lento e gli errori di timeout vengono gestiti correttamente dai client di servizio.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Imposta sia un timeout di connessione che un timeout della richiesta su qualsiasi chiamata della dipendenza del servizio e, generalmente, su qualsiasi chiamata tra i processi. Molti framework offrono funzionalità di timeout integrate, ma è necessario prestare attenzione perché alcuni sono caratterizzati da valori predefiniti infiniti o superiori a quelli accettabili per gli obiettivi dei tuoi servizi. Un valore troppo elevato riduce l'utilità del timeout perché le risorse continuano a essere consumate mentre il client attende che si verifichi il timeout. Un valore troppo basso può generare un aumento del traffico sul back-end e una maggiore latenza perché vengono ritentate troppe richieste. In alcuni casi, questo può portare a interruzioni vere e proprie perché tutte le richieste vengono ritentate.

Considera quanto segue per determinare le strategie di timeout:

- L'elaborazione delle richieste può richiedere più tempo del normale a causa del loro contenuto, di problemi nel servizio di destinazione o di un errore nella partizione della rete.
- Le richieste con contenuti troppo costosi potrebbero consumare risorse server e client non necessarie. In questo caso, forzare il timeout di queste richieste e non eseguire nuovi tentativi possono preservare le risorse. I servizi dovrebbero, inoltre, proteggersi da contenuti eccessivamente costosi con limitazioni e timeout lato server.
- Per le richieste con tempi di elaborazione eccessivamente lunghi a causa di un'interruzione del servizio è possibile forzare il timeout e, quindi, eseguire un nuovo tentativo. È necessario

considerare i costi del servizio per la richiesta e il nuovo tentativo, ma se la causa è un problema localizzato, è probabile che un nuovo tentativo non sia costoso e riduca il consumo di risorse del client. Il timeout può anche liberare risorse del server a seconda della natura del problema.

- Per le richieste il cui completamento richiede troppo tempo o per risposte non distribuite dalla rete è possibile forzare il timeout e, quindi, eseguire un nuovo tentativo. Poiché la richiesta o la risposta non è stata distribuita, viene comunque restituito un errore indipendentemente dalla durata del timeout. Il timeout in questo caso non rilascerà le risorse del server, ma le risorse del client, con il conseguente miglioramento delle prestazioni del carico di lavoro.

Sfrutta modelli di progettazione consolidati come i nuovi tentativi e interruttori per gestire normalmente i timeout e supportare l'approccio all'anticipazione degli errori (fail fast). AWS SDK e la AWS CLI consentono la configurazione dei timeout per connessioni e richieste dei nuovi tentativi con backoff esponenziale e jitter. Le funzioni AWS Lambda supportano la configurazione dei timeout e con AWS Step Functionspuoi creare interruttori a uso limitato di codice che sfruttano le integrazioni predefinite con i servizi e gli SDK AWS. AWS App Mesh Envoy fornisce funzionalità di tipo timeout e interruttore.

Passaggi dell'implementazione

- Configura i timeout per le chiamate remote dei servizi e sfrutta le funzionalità di timeout integrate o le librerie di timeout open source.
- Quando il carico di lavoro esegue chiamate con un SDK AWS, consulta la documentazione per la configurazione del timeout specifica della lingua.
  - Python
  - PHP
  - .NET
  - Ruby
  - Java
  - Go
  - Node.js
  - C++
- Quando usi SDK AWS o comandi AWS CLI nel carico di lavoro, configura i valori di timeout predefiniti impostando i valori di configurazione AWS predefiniti per `connectTimeoutInMillis` e `tlsNegotiationTimeoutInMillis`.

- Applica le opzioni della riga di comando `cli-connect-timeout` e `cli-read-timeout` per controllare i comandi AWS CLI occasionali nei servizi AWS.
- Monitora le chiamate remote dei servizi per i timeout e imposta gli allarmi sugli errori persistenti in modo da poter gestire in modo proattivo gli scenari di errore.
- Implementa le metriche CloudWatch e il rilevamento delle anomalie CloudWatch per le percentuali di errore nelle chiamate, gli obiettivi dei livelli di servizio per la latenza e i valori anomali della latenza per ottenere informazioni sulla gestione dei timeout eccessivamente restrittivi o permissivi.
- Configura i timeout per le funzioni Lambda.
- I client API Gateway devono implementare nuovi tentativi specifici durante la gestione dei timeout. API Gateway supporta un timeout di integrazione da 50 millisecondi a 29 secondi per le integrazioni downstream e non effettua nuovi tentativi quando l'integrazione richiede il timeout.
- Implementa lo schema basato sull' interruttore per evitare di effettuare chiamate remote quando si è verificato il timeout. Apri l'interruttore per evitare chiamate non riuscite e chiudi l'interruttore quando le chiamate rispondono normalmente.
- Per i carichi di lavoro basati su container, verifica le funzioni App Mesh Envoy per usare i timeout e gli interruttori integrati.
- Utilizza AWS Step Functions per creare interruttori a uso limitato di codice per le chiamate remote dei servizi, in particolare quando vengono richiamati gli SDK AWS nativi e le integrazioni Step Functions supportate per semplificare il carico di lavoro.

Risorse

Best practice correlate:

- REL05-BP03 Controllo e limitazione delle chiamate di ripetizione
- REL05-BP04 Anticipazione degli errori e limitazione delle code
- REL06-BP07 Monitoraggio del tracciamento end-to-end delle richieste attraverso il sistema

Documenti correlati:

- AWS SDK: Retries and Timeouts (SDK AWS: nuovi tentativi e timeout)
- The Amazon Builders' Library: Timeout, nuovi tentativi e backoff con jitter
- Quote Amazon API Gateway e note importanti
- AWS Command Line Interface: opzioni della riga di comando

- AWS SDK for Java 2.x: configurazione dei timeout delle API
- AWS Botocore mediante l'oggetto config e informazioni di riferimento sulla configurazione
- AWS SDK for .NET: nuovi tentativi e timeout
- AWS Lambda: configurazione delle opzioni della funzione Lambda

Esempi correlati:

- Utilizzo dello schema dell'interruttore con AWS Step Functions e Amazon DynamoDB
- Martin Fowler: CircuitBreaker

Strumenti correlati:

- AWS SDK
- AWS Lambda
- Amazon SQS
- AWS Step Functions
- AWS Command Line Interface

REL05-BP06 Utilizzo dei sistemi stateless laddove possibile

I sistemi non devono richiedere lo stato né eseguire l'offload dello stato in modo tale che, tra diverse richieste client, non vi sia alcuna dipendenza dai dati archiviati localmente su disco o in memoria. I server possono così essere sostituiti a piacimento senza compromettere la disponibilità.

Quando gli utenti o i servizi interagiscono con un'applicazione, spesso eseguono una serie di interazioni che formano una sessione. Una sessione è un dato univoco per gli utenti che persistono tra le richieste mentre utilizzano l'applicazione. Un'applicazione stateless è un'applicazione che non richiede la conoscenza delle interazioni precedenti e non memorizza le informazioni sulla sessione.

Una volta progettata per essere stateless, puoi utilizzare servizi di elaborazione serverless, come AWS Lambda o AWS Fargate.

Oltre alla sostituzione dei server, un altro vantaggio delle applicazioni stateless è la possibilità di scalare orizzontalmente, perché qualsiasi risorsa di calcolo disponibile (ad esempio istanze EC2 e funzioni AWS Lambda) può soddisfare ogni richiesta.

Vantaggi dell'adozione di questa best practice: i sistemi progettati per essere stateless sono più adattabili al dimensionamento orizzontale, rendendo possibile l'aggiunta o la rimozione di capacità in base alle fluttuazioni del traffico e della domanda. Sono inoltre intrinsecamente resistenti ai guasti e offrono flessibilità e agilità allo sviluppo delle applicazioni.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Trasforma le applicazioni in stateless. Le applicazioni stateless consentono il dimensionamento orizzontale e sono tolleranti ai guasti di un singolo nodo. Analizza e individua i componenti della tua applicazione che mantengono lo stato dell'architettura. Questo processo ti aiuta a valutare il potenziale impatto della transizione a una progettazione stateless. Un'architettura stateless separa i dati degli utenti ed esegue l'offload dei dati della sessione, offrendo la flessibilità necessaria per scalare ogni componente in modo indipendente al fine di soddisfare le diverse richieste del carico di lavoro e ottimizzare l'utilizzo delle risorse.

Passaggi dell'implementazione

- Individua e comprendi i componenti stateful dell'applicazione.
- Suddividi i dati, separando e gestendo i dati dell'utente dalla logica dell'applicazione principale.
  - Amazon Cognito è in grado di separare i dati dell'utente dal codice dell'applicazione utilizzando funzionalità quali pool di identità, pool di utenti e Amazon Cognito Sync.
  - Puoi utilizzare AWS Secrets Manager per separare i dati dell'utente archiviando i segreti in un luogo sicuro e centralizzato. Il codice dell'applicazione pertanto non dovrà più memorizzare i segreti, rendendolo più sicuro.
  - Prendi in considerazione l'utilizzo di Amazon S3 per archiviare dati non strutturati di grandi dimensioni, come immagini e documenti. L'applicazione può recuperare questi dati quando richiesto, eliminando la necessità di archiviarli in memoria.
  - Usa Amazon DynamoDB per memorizzare informazioni come i profili utente. L'applicazione può eseguire query su questi dati pressoché in tempo reale.
- Trasferisci i dati della sessione in un database, una cache o in file esterni.
  - Amazon ElastiCache, Amazon DynamoDB, Amazon Elastic File System (Amazon EFS) e Amazon MemoryDB for Redis sono esempi di servizi AWS che puoi utilizzare per eseguire l'offload dei dati della sessione.
- Progetta un'architettura stateless dopo aver identificato lo stato e i dati dell'utente che devono essere mantenuti con la tua soluzione di archiviazione preferita.

Risorse

Best practice correlate:

- REL11-BP03 Automatizzazione della riparazione a tutti i livelli

Documenti correlati:

- Amazon Builders' Library: Evitare il fallback nei sistemi distribuiti
- Amazon Builders' Library: Evitare insormontabili backlog di code
- Amazon Builders' Library: Sfide e strategie del caching
- Best practice su AWS Livello Web senza stato

REL05-BP07 Implementazione di leve di emergenza

Le leve di emergenza sono processi rapidi che possono mitigare l'impatto sulla disponibilità sul carico di lavoro.

Le leve di emergenza disabilitano, limitano o modificano il comportamento di componenti o dipendenze mediante meccanismi noti e testati. Ciò può ridurre i danni causati al carico di lavoro dall'esaurimento delle risorse dovuto ad aumenti imprevisti della domanda e l'impatto dei guasti nei componenti non critici all'interno del carico di lavoro.

Risultato desiderato: implementando le leve di emergenza, è possibile stabilire processi validi noti per garantire la disponibilità dei componenti critici nel carico di lavoro. Il carico di lavoro dovrebbe diminuire gradualmente e continuare a svolgere le sue funzioni aziendali critiche durante l'attivazione di una leva di emergenza. Per ulteriori informazioni sulla parziale riduzione delle prestazioni, consulta REL05-BP01 Implementazione della normale riduzione delle prestazioni per trasformare le dipendenze forti applicabili in dipendenze deboli.

Anti-pattern comuni:

- L'errore a livello di dipendenze non critiche influisce sulla disponibilità del carico di lavoro principale.
- Mancato test o mancata verifica del comportamento dei componenti critici durante il deterioramento delle prestazioni dei componenti non critici.
- Mancata definizione di criteri chiari e deterministici per l'attivazione o la disattivazione di una leva di emergenza.

Vantaggi dell'adozione di questa best practice: l'implementazione delle leve di emergenza può migliorare la disponibilità dei componenti critici del carico di lavoro fornendo ai risolutori processi consolidati per rispondere a picchi di domanda imprevisti o errori a livello di dipendenze non critiche.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

- Identifica i componenti critici del tuo carico di lavoro.
- Progetta e definisci l'architettura dei componenti critici del tuo carico di lavoro in modo che sia in grado di sostenere i guasti dei componenti non critici.
- Esegui i test per convalidare il comportamento dei componenti critici in caso di guasti dei componenti non critici.
- Definisci e monitora le metriche o i trigger pertinenti per avviare le procedure relative alle leve di emergenza.
- Definisci le procedure (manuali o automatiche) che includono la leva di emergenza.

Passaggi dell'implementazione

- Identifica i componenti business-critical nel tuo carico di lavoro.
  - Ogni componente tecnico del carico di lavoro deve essere mappato alla funzione aziendale pertinente e classificato come critico o non critico. Per esempi di funzionalità critiche e non critiche in Amazon, consulta Any Day Can Be Prime Day: How Amazon.com Search Uses Chaos Engineering to Handle Over 84K Requests Per Second (informazioni in lingua inglese).
  - Si tratta di una decisione sia tecnica che aziendale e varia in base all'organizzazione e al carico di lavoro.
- Progetta e definisci l'architettura dei componenti critici del tuo carico di lavoro in modo che sia in grado di sostenere i guasti dei componenti non critici.
  - Durante l'analisi delle dipendenze, valuta tutte le potenziali modalità di guasto e verifica che i meccanismi basati su leve di emergenza forniscano le funzionalità critiche ai componenti a valle.
- Esegui i test per convalidare il comportamento dei componenti critici durante l'attivazione delle leve di emergenza.
  - Evita il comportamento bimodale. Per maggiori dettagli, consulta REL11-BP05 Utilizzo della stabilità statica per evitare un comportamento bimodale.
- Definisci, monitora e attiva gli avvisi per le metriche pertinenti per avviare la procedura relative alla leva di emergenza.

  - L'individuazione delle metriche da monitorare dipende dal carico di lavoro. Alcuni esempi di metrica sono la latenza o il numero di richieste non riuscite nei confronti di una dipendenza.
- Definisci le procedure (manuali o automatiche) che includono la leva di emergenza.
  - Ciò può includere meccanismi come la riduzione del carico, le richieste di limitazione della larghezza di banda della rete (throttling) o l'implementazione di una parziale riduzione delle prestazioni.

Risorse

Best practice correlate:

- REL05-BP01 Implementazione della normale riduzione delle prestazioni per trasformare le dipendenze forti applicabili in dipendenze deboli
- REL05-BP02 Richieste di limitazione (della larghezza di banda della rete)
- REL11-BP05 Utilizzo della stabilità statica per evitare un comportamento bimodale

Documenti correlati:

- Automazione di implementazioni pratiche e sicure
- Any Day Can Be Prime Day: How Amazon.com Search Uses Chaos Engineering to Handle Over 84K Requests Per Second (informazioni in lingua inglese)

Video correlati:

- AWS re:Invent 2020: Reliability, consistency, and confidence through immutability

Gestione delle modifiche

Domande

- REL 6. Come monitorare le risorse del carico di lavoro?
- REL 7. Come si progetta il carico di lavoro per adattarsi ai cambiamenti della domanda?
- REL 8. In che modo implementare le modifiche?

REL 6. Come monitorare le risorse del carico di lavoro?

I log e i parametri sono strumenti molto efficaci per ottenere informazioni sullo stato del tuo carico di lavoro. È possibile configurare il carico di lavoro in modo da monitorare i log e i parametri e inviare notifiche quando vengono superate le soglie o si verificano eventi significativi. Il monitoraggio permette al carico di lavoro di riconoscere quando vengono superate le soglie di prestazioni basse o si verificano errori, in modo che possa essere ripristinato automaticamente di rimando.

Best practice

- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL06-BP02 Definizione e calcolo dei parametri (aggregazione)
- REL06-BP03 Invio di notifiche (elaborazione e avvisi in tempo reale)
- REL06-BP04 Automatizzazione delle risposte (elaborazione e avvisi in tempo reale)
- REL06-BP05 Analisi
- REL06-BP06 Esecuzione di revisioni periodiche
- REL06-BP07 Monitoraggio del tracciamento end-to-end delle richieste attraverso il sistema

REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)

monitora i componenti del carico di lavoro con Amazon CloudWatch o con strumenti di terze parti. Monitora i servizi AWS con il pannello di controllo AWS Health.

Occorre monitorare tutti i componenti del carico di lavoro, inclusi front-end, logica aziendale e livelli di storage. Definisci i parametri chiave e come estrarli dai registri, se necessario, e imposta soglie per l'attivazione degli eventi di allarme corrispondenti. Assicurati che i parametri siano pertinenti agli indicatori chiave di prestazione (KPI) del tuo carico di lavoro e utilizza i parametri e i registri per identificare i primi segnali di degrado del servizio. Ad esempio, un parametro legato ai risultati aziendali, come il numero di ordini elaborati con successo al minuto, può indicare problemi di carico di lavoro più rapidamente di un parametro tecnico, come l'utilizzo della CPU. Utilizza il pannello di controllo AWS Health per una visualizzazione personalizzata delle prestazioni e della disponibilità dei servizi AWS sottostanti alle risorse AWS.

Il monitoraggio nel cloud offre nuove opportunità. La maggior parte dei provider cloud ha sviluppato hook personalizzabili e può fornire approfondimenti per aiutarti a monitorare più livelli del carico di lavoro. I servizi AWS come Amazon CloudWatch applicano algoritmi statistici e di apprendimento automatico per analizzare continuamente i parametri di sistemi e applicazioni, determinare le normali linee di base e far emergere le anomalie con un intervento minimo da parte dell'utente. Gli algoritmi

di rilevamento delle anomalie tengono conto della stagionalità e delle variazioni di tendenza dei parametri.

AWS mette a disposizione una grande quantità di informazioni di monitoraggio e di registro che possono essere utilizzate per definire parametri specifici per i carichi di lavoro, processi di variazione della domanda e per l'adozione di tecniche di apprendimento automatico indipendentemente dalle competenze di ML.

Inoltre, monitora tutti gli endpoint esterni per avere la certezza che siano indipendenti dall'implementazione di base. Questo monitoraggio attivo può essere effettuato con transazioni sintetiche (talvolta indicate come canary utente,ma da non confondere con le implementazioni canary) che eseguono periodicamente una serie di attività comuni che corrispondono alle azioni eseguite dai client del carico di lavoro. Mantieni queste attività di breve durata e assicurati di non sovraccaricare il carico di lavoro durante il test. Amazon CloudWatch Synthetics ti consente di creare canary sintetici per monitorare gli endpoint e le API. Puoi anche combinare i nodi client sintetici Canary con la console AWS X-Ray per individuare quali Canary sintetiche stanno riscontrando problemi con errori, guasti o velocità di throttling per l'intervallo di tempo selezionato.

Risultato desiderato:

raccogliere e utilizzare i parametri critici di tutti i componenti del carico di lavoro per garantire l'affidabilità del carico di lavoro e un'esperienza utente ottimale. Rilevare che un carico di lavoro non sta raggiungendo i risultati aziendali consente di dichiarare rapidamente un disastro e di riprendersi da un incidente.

Anti-pattern comuni:

- Solo monitoraggio delle interfacce esterne per il carico di lavoro.
- Non generare parametri specifici per il carico di lavoro e affidati solo ai parametri forniti dai servizi AWS utilizzati dal carico di lavoro.
- Utilizzare solo parametri tecnici nel carico di lavoro e non monitorare i parametri relativi agli indicatori chiave di prestazione (KPI) non tecnici a cui il carico di lavoro contribuisce.
- Affidarsi al traffico di produzione e a semplici controlli di integrità per monitorare e valutare lo stato del carico di lavoro.

Vantaggi dell'adozione di questa best practice: il monitoraggio a tutti i livelli del carico di lavoro consente di prevedere e risolvere più rapidamente i problemi dei componenti che costituiscono il carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: Alta

Guida all'implementazione

1. Abilitazione della registrazione ove disponibile. I dati di monitoraggio devono essere ottenuti da tutti i componenti dei carichi di lavoro. Attiva ulteriori registri, come i registri di accesso S3, e abilita il carico di lavoro per registrare i dati specifici del carico di lavoro. Raccogli i parametri per le medie di CPU, I/O di rete e I/O su disco da servizi come Amazon ECS, Amazon EKS, Amazon EC2, Elastic Load Balancing, AWS Auto Scaling ed Amazon EMR. Consulta Servizi AWS che pubblicano parametri CloudWatch Servizi AWS che pubblicano parametri su CloudWatch.
2. Esamina tutti i parametri predefiniti ed esplora eventuali lacune nella raccolta dei dati. Tutti i servizi generano parametri predefiniti. La raccolta di parametri predefiniti consente di comprendere meglio le dipendenze tra i componenti del carico di lavoro e il modo in cui l'affidabilità e le prestazioni dei componenti influiscono sul carico di lavoro. Puoi anche creare e pubblicare parametri propri affinché CloudWatch utilizzi la AWS CLI o un'API. Questo
3. valuta tutti i parametri per decidere quelli a cui inviare avvisi per ogni servizio AWS nel carico di lavoro. Puoi scegliere di selezionare un sottoinsieme di parametri che hanno un impatto importante sull'affidabilità del carico di lavoro. La focalizzazione su soglie e parametri critici consente di affinare il numero di avvisi informativi e può contribuire a ridurre al minimo i falsi positivi.
4. Definisci gli avvisi e il processo di recupero del carico di lavoro dopo l'attivazione dell'avviso. La definizione degli avvisi consente di notificare, intensificare e seguire rapidamente le fasi necessarie per il ripristino da un incidente e il rispetto dell'obiettivo di tempo di ripristino (RTO) prescritto. Puoi utilizzare avvisi Amazon CloudWatch per invocare flussi di lavoro automatici e avviare procedure di ripristino in base a soglie definite.
5. Esplora l'uso di transazioni sintetiche per raccogliere dati rilevanti sullo stato dei carichi di lavoro. Il monitoraggio sintetico segue gli stessi percorsi ed esegue le stesse azioni di un cliente, il che consente di verificare continuamente l'esperienza del cliente anche quando non c'è traffico di clienti sui carichi di lavoro. Utilizzando le transazioni sintetiche,puoi individuare i problemi prima dei clienti.

Risorse

Best practice correlate:

- REL11-BP03 Automatizzazione della riparazione a tutti i livelli

Documenti correlati:

- Getting started with your AWS Health Dashboard – Your account health (Nozioni di base su AWS HealthDashboard: stato del tuo account)
- Servizi AWS che pubblicano parametri CloudWatch
- Log di accesso per Network Load Balancer
- Log di accesso per Application Load Balancer
- Accesso a Amazon CloudWatch Logs per AWS Lambda
- Registrazione delle richieste con registrazione dell'accesso al server Amazon S3
- Abilita i log di accesso per Classic Load Balancer
- Esportazione di dati di registro in Amazon S3
- Installazione dell'agente CloudWatch su un'istanza Amazon EC2
- Pubblicazione di parametri personalizzati
- Utilizzo dei pannelli di controllo Amazon CloudWatch
- Utilizzare i parametri Amazon CloudWatch
- Utilizzo di Canary (Amazon CloudWatch Synthetics)
- Cosa sono i Amazon CloudWatch Logs?

Guide per l'utente:

- Creazione di un trail
- Monitoraggio dei parametri di memoria e del disco per le istanze Amazon EC2 Linux
- Utilizzo di CloudWatch Logs con istanze di container
- Log di flusso VPC
- Che cos'è Amazon DevOps Guru?
- Che cos'è AWS X-Ray?

Blog correlati:

- Effettuare il debug con Amazon CloudWatch Synthetics e AWS X-Ray

Esempi e workshop correlati:

- AWS Well-Architected Labs: Operational Excellence - Dependency Monitoring (Laboratori ben strutturati AWS: Eccellenza operativa - Monitoraggio delle dipendenze)
- The Amazon Builders' Library: Dotazione dei sistemi distribuiti per la visibilità operativa

- Workshop sull'osservabilità

REL06-BP02 Definizione e calcolo dei parametri (aggregazione)

Archivia i dati di registro e applica i filtri, laddove necessari, per calcolare i parametri, ad esempio i conteggi di un evento di registro specifico o la latenza calcolata dai timestamp del registro eventi.

Amazon CloudWatch e Amazon S3 fungono da principali livelli di aggregazione e storage. Per alcuni servizi, come AWS Auto Scaling e Elastic Load Balancing, i parametri predefiniti vengono forniti per impostazione predefinita per il carico della CPU o la latenza media delle richieste in un cluster o in un'istanza. Per i servizi di streaming, come i registri di flusso VPC e AWS CloudTrail, i dati degli eventi vengono inoltrati a CloudWatch Logs ed è necessario definire e applicare filtri di parametri per estrarre i parametri dai dati dell'evento. In questo modo vengono forniti dati di serie temporali, che possono fungere da input per gli allarmi CloudWatch definiti dall'utente per attivare gli avvisi.

Livello di rischio associato se questa best practice non fosse adottata: Alta

Guida all'implementazione

- Aggregazione: definisci e calcola i parametri. Archivia i dati di log e applica filtri, se necessario, per calcolare i parametri, ad esempio i conteggi di un evento di log specifico o la latenza calcolata dai timestamp degli eventi di log
  - I filtri dei parametri definiscono i termini e i modelli da ricercare nei dati di registro inviati a CloudWatch Logs. CloudWatch Logs utilizza questi filtri di parametri per trasformare i dati di registro in parametri CloudWatch numerici che è possibile rappresentare su un grafico o un avviso.
    - Ricerca e filtraggio dei dati di log
  - Utilizza una terza parte affidabile per aggregare i registri.
    - Segui le istruzioni che ti vengono fornite dalle terze parti. La maggior parte dei prodotti di terze parti si integra con CloudWatch e Amazon S3.
  - Alcuni servizi AWS possono pubblicare registri direttamente in Amazon S3. Se il requisito principale per i registri è l'archiviazione in Amazon S3, si può facilmente fare in modo che il servizio che produce i registri li invii direttamente a Amazon S3, senza dover creare un'infrastruttura aggiuntiva.
    - Invio di registri direttamente a Amazon S3

Risorse

Documenti correlati:

- Query di esempio di Amazon CloudWatch Logs Insights
- Effettuare il debug con Amazon CloudWatch Synthetics e AWS X-Ray
- One Observability Workshop
- Ricerca e filtraggio dei dati di log
- Invio di registri direttamente a Amazon S3
- The Amazon Builders' Library: Dotazione dei sistemi distribuiti per la visibilità operativa

REL06-BP03 Invio di notifiche (elaborazione e avvisi in tempo reale)

Quando le organizzazioni rilevano potenziali problemi, inviano notifiche e avvisi in tempo reale ai team e ai sistemi appropriati per rispondere rapidamente ed efficacemente alle difficoltà.

Risultato desiderato: è possibile rispondere rapidamente agli eventi operativi attraverso la configurazione di allarmi pertinenti in base ai parametri del servizio e dell'applicazione. Quando la soglia degli allarmi viene superata, i team e i sistemi appropriati vengono informati in modo che possano risolvere i problemi sottostanti.

Anti-pattern comuni:

- Configuri gli allarmi con una soglia eccessivamente alta, con conseguente mancato invio di notifiche importanti.
- Configuri gli allarmi con una soglia troppo bassa, con il risultato che gli avvisi importanti non vengono presi in considerazione a causa del numero eccessivo di notifiche generate.
- Non aggiorni gli allarmi e la relativa soglia quando cambia l'utilizzo.
- Per gli allarmi gestiti meglio tramite le azioni automatizzate, l'invio della notifica ai team anziché l'attivazione dell'azione automatizzata comporta la generazione di un numero eccessivo di notifiche.

Vantaggi dell'adozione di questa best practice: l'invio di notifiche e avvisi in tempo reale ai team e ai sistemi appropriati consente di individuare tempestivamente i problemi e di rispondere rapidamente agli incidenti operativi.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

I carichi di lavoro devono essere dotati di sistemi di elaborazione e allarme in tempo reale per migliorare l'identificazione dei problemi che possono influire sulla disponibilità dell'applicazione e fungere da trigger per la risposta automatizzata. Le organizzazioni possono eseguire un sistema di elaborazione e allarme in tempo reale creando avvisi con parametri definiti in modo da ricevere le notifiche ogni volta che si verificano eventi significativi o un parametro supera una determinata soglia.

Amazon CloudWatch ti permette di creare allarmi compositi e di parametri utilizzando gli allarmi CloudWatch basati su soglie statiche, rilevamento di anomalie e altri criteri. Per maggiori dettagli sui tipi di allarmi che puoi configurare utilizzando CloudWatch, consulta la sezione allarmi della documentazione CloudWatch.

Puoi creare per i tuoi team visualizzazioni personalizzate dei parametri e degli avvisi delle risorse AWS utilizzando le dashboard CloudWatch. Le home page personalizzabili nella console di CloudWatch consentono di monitorare le risorse di più regioni in un'unica visualizzazione.

Gli allarmi possono eseguire una o più azioni, come inviare una notifica a un argomento Amazon SNS, eseguendo un'azione su Amazon EC2 o un'azione su Amazon EC2 Auto Scaling oppure creando un OpsItem o a in AWS Systems Manager.

Amazon CloudWatch utilizza Amazon SNS per inviare le notifiche quando l'allarme cambia stato, con la distribuzione dei messaggi degli editori (produttori) agli abbonati (consumatori). Per maggiori dettagli sull'impostazione delle notifiche Amazon SNS, consulta Configurazione di Amazon SNS.

CloudWatch invia EventBridge della sicurezza ogni volta che un allarme CloudWatch viene creato, aggiornato, eliminato o cambia stato. Puoi usare EventBridge con questi eventi per creare le regole che eseguono le azioni, come avvisare ogni volta che lo stato di un allarme cambia o attivare automaticamente gli eventi nel tuo account tramite l'automazione Systems Manager.

Quando si usa EventBridge rispetto ad Amazon SNS?

EventBridge e Amazon SNS possono entrambi essere utilizzati per sviluppare applicazioni basate su eventi e la scelta dipende dalle tue esigenze specifiche.

Amazon EventBridge è consigliato quando desideri creare un'applicazione che reagisca agli eventi delle tue applicazioni, delle applicazioni SaaS e dei servizi AWS. EventBridge è l'unico servizio basato su eventi che si integra direttamente con i partner SaaS di terze parti. EventBridge inoltre acquisisce automaticamente eventi da oltre 200 servizi AWS senza richiedere agli sviluppatori di creare risorse negli account.

EventBridge utilizza una struttura definita basata su JSON per gli eventi e consente di creare regole applicate all'intero corpo dell'evento per selezionare gli eventi da inoltrare alle destinazioni. EventBridge attualmente supporta oltre 20 servizi AWS come destinazioni, tra cui AWS Lambda, Amazon SQS, Amazon SNS, Amazon Kinesis Data Streamse Amazon Data Firehose.

Amazon SNS è consigliato per le applicazioni che richiedono un fan-out elevato (migliaia o milioni di endpoint). Di solito i clienti utilizzano Amazon SNS come destinazione della regola per filtrare gli eventi di cui hanno bisogno e sottoporli al fan-out su più endpoint.

I messaggi non sono strutturati e possono essere in qualsiasi formato. Amazon SNS supporta l'inoltro dei messaggi a sei diversi tipi di destinazioni, tra cui Lambda, Amazon SQS, endpoint HTTP/S, SMS, push mobile ed e-mail. La latenza tipica di Amazon SNS è inferiore a 30 millisecondi. Un'ampia gamma di servizi AWS invia i messaggi Amazon SNS definendo la configurazione appropriata (più di 30, inclusi Amazon EC2, Amazon S3e Amazon RDS).

Passaggi dell'implementazione

1. Crea un allarme usando gli avvisi Amazon CloudWatch.
    a. Un allarme di parametri monitora un singolo parametro CloudWatch o un'espressione dipendente dai parametri CloudWatch. L'allarme avvia una o più azioni in base al valore del parametro o dell'espressione rispetto a una soglia, per un determinato numero di intervalli di tempo. L'azione può consistere nell'inviare una notifica a un argomento Amazon SNS, eseguendo un'azione su Amazon EC2 o un'azione su Amazon EC2 Auto Scaling oppure creando un OpsItem o a in AWS Systems Manager.
    b. Un allarme composito è costituito da un'espressione di regola che considera le condizioni di altri allarmi che hai creato. L'allarme composito entra in stato di allarme solo se tutte le condizioni della regola sono soddisfatte. Gli allarmi specificati nell'espressione di regola di un allarme composito possono includere allarmi di parametri e allarmi compositi aggiuntivi. Gli allarmi compositi possono inviare notifiche Amazon SNS quando il loro stato cambia e possono creare Systems Manager OpsItems o incidenti quando entrano nello stato di allarme, ma non possono eseguire azioni Amazon EC2 o Auto Scaling.
2. Configura le notifiche Amazon SNS. Quando si crea un allarme CloudWatch, è possibile includere un argomento Amazon SNS per inviare una notifica quando l'allarme cambia stato.
3. Crea regole in EventBridge che corrisponde agli allarmi CloudWatch specificati. Ogni regola supporta più destinazioni, incluse le funzioni Lambda. Ad esempio, è possibile definire un allarme che si attiva quando lo spazio disponibile su disco si sta esaurendo e che esegue una

funzione Lambda tramite una regola EventBridge per ripulire lo spazio. Per maggiori dettagli sulle destinazioni EventBridge, consulta Destinazioni EventBridge.

Risorse

Best practice Well-Architected correlate:

- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL06-BP02 Definizione e calcolo dei parametri (aggregazione)
- REL12-BP01 Utilizzo dei playbook per analizzare gli errori

Documenti correlati:

- Amazon CloudWatch
- CloudWatch Logs insights
- Utilizzo degli allarmi di Amazon CloudWatch
- Utilizzo dei pannelli di controllo Amazon CloudWatch
- Utilizzare i parametri Amazon CloudWatch
- Setting up Amazon SNS notifications
- il rilevamento delle anomalie CloudWatch
- Protezione dei dati CloudWatch Logs
- Amazon EventBridge
- Amazon Simple Notification Service

Video correlati:

- Video sull'osservabilità di reinvent 2022
- AWS re:Invent 2022 - Observability best practices at Amazon

Esempi correlati:

- One Observability Workshop
- Amazon EventBridge to AWS Lambda with feedback control by Amazon CloudWatch Alarms

REL06-BP04 Automatizzazione delle risposte (elaborazione e avvisi in tempo reale)

utilizza l'automazione per agire quando viene rilevato un evento; ad esempio, per sostituire i componenti guasti.

L'elaborazione automatizzata in tempo reale degli allarmi è implementata in modo che i sistemi possano effettuare azioni correttive rapide e tentare di prevenire guasti o danni al servizio quando vengono attivati gli allarmi. Le risposte automatiche agli allarmi potrebbero includere la sostituzione dei componenti guasti, la regolazione della capacità di calcolo, il reindirizzamento del traffico verso host integri, zone di disponibilità o altre regioni e la notifica agli operatori.

Risultato desiderato: vengono identificati gli allarmi in tempo reale e viene impostata l'elaborazione automatizzata degli allarmi per richiamare le azioni appropriate per mantenere gli obiettivi dei livelli di servizio e gli accordi sul livello di servizio (SLA). L'automazione può interessare un ambito che va dalle attività di autoriparazione dei singoli componenti al failover dell'intero sito.

Anti-pattern comuni:

- Non disporre di un inventario o un catalogo dettagliato dei principali allarmi in tempo reale.
- Nessuna risposta automatica in caso di allarmi critici (ad esempio, quando le risorse di calcolo stanno per esaurirsi, viene implementato il dimensionamento automatico).
- Azioni di risposta agli allarmi contraddittorie.
- Nessuna procedura operativa standard (SOP) da seguire per gli operatori quando ricevono notifiche di avviso.
- Non monitorare le modifiche apportate alla configurazione, poiché le modifiche della configurazione non rilevate possono causare tempi di inattività per i carichi di lavoro.
- Non avere una strategia per annullare le modifiche involontarie alla configurazione.

Vantaggi dell'adozione di questa best practice: l'automazione dell'elaborazione degli allarmi può migliorare la resilienza del sistema. Il sistema implementa automaticamente azioni correttive, riducendo le attività manuali che possono comportare interventi umani soggetti a errori. L'operatività del carico di lavoro soddisfa gli obiettivi di disponibilità e riduce le interruzioni del servizio.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Per gestire in modo efficiente gli avvisi e automatizzarne la risposta, classifica gli avvisi in base alla loro criticità e al loro impatto, documenta le procedure di risposta e pianifica le risposte prima di classificare le attività.

Identifica le attività che richiedono azioni specifiche (spesso dettagliate nei runbook) ed esamina tutti i runbook e i playbook per determinare quali attività possono essere automatizzate. Se è possibile definire delle azioni, significa che esse spesso possono essere automatizzate. Se le azioni non possono essere automatizzate, documenta le fasi manuali in una procedura operativa standard (SOP) e forma gli operatori su tali procedure. Continua ad analizzare dettagliatamente i processi manuali alla ricerca di opportunità di automazione in cui puoi stabilire e mantenere un piano per automatizzare le risposte agli avvisi.

Passaggi dell'implementazione

1. Crea un inventario degli allarmi: per ottenere un elenco di tutti gli allarmi, nella AWS CLI puoi utilizzare il comando Amazon CloudWatch `describe-alarms`. A seconda del numero di allarmi configurati, potrebbe essere necessario utilizzare la paginazione per recuperare un sottoinsieme di allarmi per ogni chiamata o, in alternativa, è possibile utilizzare AWS SDK per recuperare gli allarmi mediante una chiamata API.
2. Documenta tutte le azioni degli allarmi: aggiorna un runbook con tutti gli allarmi e le relative azioni, indipendentemente dal fatto che siano manuali o automatiche. AWS Systems Manager fornisce runbook predefiniti. Per ulteriori informazioni sui runbook, consulta Working with runbooks. Per informazioni dettagliate su come visualizzare il contenuto del runbook, consulta View runbook content.
3. Configura e gestisci le azioni associate agli allarmi: per tutti gli allarmi che richiedono un'azione, specifica l'azione automatizzata mediante CloudWatch SDK. Ad esempio, puoi modificare automaticamente lo stato delle tue istanze Amazon EC2 in base a un allarme CloudWatch creando e abilitando o disabilitando le azioni associate a un allarme.

   Puoi anche utilizzare Amazon EventBridge per rispondere automaticamente agli eventi di sistema, come problemi di disponibilità delle applicazioni o modifiche delle risorse. Puoi creare regole per indicare quali eventi ti interessano e le azioni da eseguire quando un evento soddisfa una regola. Le azioni che possono essere avviate automaticamente includono il richiamo di una funzione AWS Lambda, il richiamo della funzionalità Amazon EC2 `Run Command`, l'inoltro dell'evento a Amazon Kinesis Data Streams e la visualizzazione del comando Automate Amazon EC2 mediante EventBridge.

4. Procedure operative standard (SOP): in base ai componenti dell'applicazione, AWS Resilience Hub consiglia più modelli SOP. È possibile utilizzare queste SOP per documentare tutti i processi che un operatore deve seguire nel caso in cui venga generato un avviso. Puoi anche creare una SOP basata su raccomandazioni Resilience Hub, laddove sia necessaria un'applicazione Resilience Hubcon una policy di resilienza associata, nonché una valutazione cronologica della resilienza rispetto a tale applicazione. Le raccomandazioni per la SOP sono prodotte dalla valutazione della resilienza.

   Resilience Hub in combinazione con Systems Manager consente di automatizzare le fasi delle SOP fornendo una serie di documenti SSM che è possibile utilizzare come base per tali SOP. Ad esempio, Resilience Hub può consigliare una SOP per aggiungere spazio su disco in base a un documento SSM di automazione esistente.
5. Esegui azioni automatizzate utilizzando Amazon DevOps Guru: puoi utilizzare Amazon DevOps Guru per monitorare automaticamente le risorse dell'applicazione per rilevare comportamenti anomali e fornire raccomandazioni mirate per accelerare i tempi di identificazione e riparazione dei problemi. Con DevOps Guru, puoi monitorare flussi di dati operativi quasi in tempo reale da più origini, tra cui metriche Amazon CloudWatch, AWS Config, AWS CloudFormation e AWS X-Ray. È inoltre possibile utilizzare DevOps Guru per creare automaticamente OpsItems in OpsCenter e inviare eventi a EventBridge per un'automazione aggiuntiva.

Risorse

Best practice correlate:

- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL06-BP02 Definizione e calcolo dei parametri (aggregazione)
- REL06-BP03 Invio di notifiche (elaborazione e avvisi in tempo reale)
- REL08-BP01 Utilizzo di runbook per attività standard come l'implementazione

Documenti correlati:

- AWS Systems Manager Automation
- Creating an EventBridge Rule That Triggers on an Event from an AWS Resource
- One Observability Workshop
- The Amazon Builders' Library: Dotazione dei sistemi distribuiti per la visibilità operativa
- What is Amazon DevOps Guru?

- Gestione dei documenti di automazione (playbook)

Video correlati:

- AWS re:Invent 2022 - Best practice di visibilità in Amazon
- AWS re:Invent 2020: Automate anything with AWS Systems Manager
- Introduction to AWS Resilience Hub
- Create Custom Ticket Systems for Amazon DevOps Guru Notifications
- Enable Multi-Account Insight Aggregation with Amazon DevOps Guru

Esempi correlati:

- Workshop sull'affidabilità
- Workshop su Amazon CloudWatch e Systems Manager

REL06-BP05 Analisi

raccogli i file di log e le cronologie dei parametri e analizzali per ottenere informazioni più ampie sulle tendenze e sui carichi di lavoro.

Amazon CloudWatch Logs Insights supporta un linguaggio di query semplice ma potente che puoi utilizzare per analizzare i dati di log. Amazon CloudWatch Logs supporta anche le sottoscrizioni che consentono ai dati di fluire in modo ottimale verso Amazon S3, dove puoi utilizzare o Amazon Athena per eseguire query sui dati. Supporta, inoltre, le query su un'ampia gamma di formati. Consulta SerDe e formati di dati supportati nella Guida per l'utente Amazon Athena per ulteriori informazioni. Per l'analisi di enormi set di file di log, puoi eseguire un cluster Amazon EMR per effettuare analisi con capacità nell'ordine dei petabyte.

Esistono numerosi strumenti forniti da Partner AWS e terze parti che consentono aggregazione, elaborazione, archiviazione e analisi. Questi strumenti includono New Relic, Splunk, Loggly, Logstash, CloudHealth e Nagios. Tuttavia, la generazione esterna di log di sistema e applicazioni è univoca per ciascun provider di servizi cloud e spesso per ciascun servizio.

Una parte spesso trascurata del processo di monitoraggio è la gestione dei dati. È necessario determinare i requisiti di conservazione per il monitoraggio dei dati, quindi applicare le policy del ciclo di vita di conseguenza. Amazon S3 supporta la gestione del ciclo di vita a livello di bucket S3.

Questa gestione del ciclo di vita può essere applicata in modo diverso ai diversi percorsi nel bucket. Verso la fine del ciclo di vita è possibile trasferire i dati su Amazon S3 Glacier per l'archiviazione a lungo termine fino alla scadenza, al termine del periodo di conservazione. La classe di storage S3 Intelligent-Tiering è progettata per ottimizzare i costi trasferendo automaticamente i dati nel livello di accesso più conveniente, senza impatto sulle prestazioni o sovraccarico operativo.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

- Gli approfondimenti CloudWatch Logs consentono di cercare e analizzare in modo interattivo i dati di registro in Amazon CloudWatch Logs.
  - Analisi dei dati di registro con gli approfondimenti CloudWatch Logs
  - Query di esempio di Amazon CloudWatch Logs Insights
- Utilizza Amazon CloudWatch Logs per inviare registri a Amazon S3 dove puoi utilizzare Amazon Athena per le query dei dati.
  - Come faccio ad analizzare i miei registri di accesso al server Amazon S3 utilizzando Athena?
    - Crea una policy del ciclo di vita di S3 per il bucket dei log di accesso al server. Configura la policy del ciclo di vita per rimuovere periodicamente i file di log. In questo modo si riduce la quantità di dati che Athena deve analizzare per ogni query.
      - Come faccio a creare una policy del ciclo di vita per un bucket S3?

Risorse

Documenti correlati:

- Query di esempio di Amazon CloudWatch Logs Insights
- Analisi dei dati di registro con gli approfondimenti CloudWatch Logs
- Effettuare il debug con Amazon CloudWatch Synthetics e AWS X-Ray
- Come faccio a creare una policy del ciclo di vita per un bucket S3?
- Come faccio ad analizzare i miei registri di accesso al server Amazon S3 utilizzando Athena?
- One Observability Workshop
- The Amazon Builders' Library: Dotazione dei sistemi distribuiti per la visibilità operativa

REL06-BP06 Esecuzione di revisioni periodiche

Esegui verifiche frequenti delle modalità di implementazione del monitoraggio del carico di lavoro e aggiornalo in base a eventi e modifiche significativi.

Il monitoraggio efficace è basato su parametri aziendali chiave. Assicurati che questi parametri siano presenti nel carico di lavoro man mano che le priorità aziendali cambiano.

L'audit del monitoraggio consente di sapere quando un'applicazione sta raggiungendo gli obiettivi di disponibilità. L'analisi delle cause principali richiede la capacità di scoprire cosa è successo in caso di errori. AWS consente di monitorare lo stato dei tuoi servizi durante un incidente:

- Amazon CloudWatch Logs: è possibile archiviare i log in questo servizio e controllarne i contenuti.
- Amazon CloudWatch Logs Insights: è un servizio completamente gestito che consente di eseguire analisi di registri di grandi dimensioni in pochi secondi. Offre query e visualizzazioni rapide e interattive.
- AWS Config: è possibile vedere quale infrastruttura AWS era in uso in momenti differenti.
- AWS CloudTrail: è possibile vedere quali API AWS sono state richiamate, a che ora e da quale principale.

In AWS, conduciamo meeting settimanali per esaminare le prestazioni operative e condividere quanto appreso tra i team. Dato l'elevato numero di team presenti in AWS, abbiamo creato La ruota per scegliere casualmente un carico di lavoro da esaminare. Stabilire una cadenza regolare per le revisioni delle prestazioni operative e la condivisione delle conoscenze migliora la capacità di ottenere prestazioni più elevate dai team operativi.

Anti-pattern comuni:

- Raccolta dei soli parametri predefiniti.
- Impostazione di una strategia di monitoraggio senza alcuna revisione.
- Nessuna discussione sul monitoraggio quando vengono distribuite modifiche importanti.

Vantaggi dell'adozione di questa best practice: la verifica periodica del monitoraggio consente di prevedere potenziali problemi, invece di rispondere alle notifiche quando un problema previsto si verifica effettivamente.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

- Crea più pannelli di controllo per il carico di lavoro. È necessario disporre di un pannello di controllo di primo livello contenente i parametri aziendali chiave, nonché i parametri tecnici che hai identificato come i più rilevanti per lo stato previsto del carico di lavoro al variare dell'utilizzo. È inoltre importante disporre di pannelli di controllo per vari livelli di applicazione e dipendenze che è possibile ispezionare.
  - Utilizzo dei pannelli di controllo Amazon CloudWatch
- Pianifica ed effettua revisioni periodiche dei pannelli di controllo del carico di lavoro. Effettua un'ispezione regolare dei pannelli di controllo. La frequenza può essere diversa a seconda di quanto l'ispezione sia approfondita.
  - Ispeziona l'andamento nei parametri. Confronta i valori dei parametri con i valori storici per vedere se ci sono tendenze che potrebbero suggerire l'esame di un particolare aspetto. Riportiamo alcuni esempi: aumento della latenza, riduzione della funzione aziendale primaria e aumento delle risposte all'errore.
  - Identificazione di outlier/anomalie nei parametri. Le medie o mediane possono nascondere outlier e anomalie. Osserva i valori più alti e più bassi nell'intervallo di tempo e analizza le cause dei risultati estremi. Man mano che continui a eliminare tali cause, la riduzione del numero di valori estremi ti consente di continuare a migliorare la coerenza delle prestazioni del carico di lavoro.
  - Ricerca di bruschi cambiamenti nel comportamento. Un cambiamento repentino della quantità o della direzione di un parametro può indicare un cambiamento nell'applicazione o fattori esterni che potrebbero richiedere l'aggiunta di ulteriori parametri da monitorare.

Risorse

Documenti correlati:

- Query di esempio di Amazon CloudWatch Logs Insights
- Effettuare il debug con Amazon CloudWatch Synthetics e AWS X-Ray
- One Observability Workshop
- The Amazon Builders' Library: Dotazione dei sistemi distribuiti per la visibilità operativa
- Utilizzo dei pannelli di controllo Amazon CloudWatch

REL06-BP07 Monitoraggio del tracciamento end-to-end delle richieste attraverso il sistema

Tieni traccia delle richieste durante l'elaborazione dei componenti del servizio in modo che i team del prodotto possano analizzare i problemi, semplificarne il debug e migliorare le prestazioni.

Risultato desiderato: I carichi di lavoro con tracciabilità completa di tutti i componenti sono caratterizzati da processi di debug più semplici e ciò migliora il tempo medio di risoluzione (MTTR) degli errori e la latenza grazie alla semplificazione dell'individuazione delle cause principali. La tracciabilità end-to-end riduce il tempo necessario per individuare i componenti interessati e approfondire in dettaglio le cause principali degli errori o della latenza.

Anti-pattern comuni:

- Il tracciamento viene utilizzato per alcuni componenti ma non per tutti. Ad esempio, senza il tracciamento AWS Lambda, i team potrebbero non avere una chiara comprensione della latenza causata dagli avviamenti a freddo in un periodo di picco del carico di lavoro.
- I canary Synthetics o le metriche RUM (Real-User Monitoring) non sono configurati con il tracciamento. Senza canary o metriche RUM, la telemetria delle interazioni dei clienti viene omessa dall'analisi dei tracciamenti e ciò rende incompleto il profilo delle prestazioni.
- I carichi di lavoro ibridi includono strumenti di tracciamento nativi del cloud e di terze parti, ma non sono state prese misure specifiche per selezionare e integrare completamente un'unica soluzione di tracciamento. In base alla soluzione di tracciamento scelta, gli SDK di tracciamento nativi del cloud devono essere utilizzati per instrumentare i componenti non nativi del cloud oppure è necessario configurare strumenti di terze parti per acquisire i dati telemetrici delle tracce nativi del cloud.

Vantaggi dell'adozione di questa best practice: Quando vengono avvisati della presenza di problemi, i team di sviluppo possono visualizzare un quadro completo delle interazioni tra i componenti del sistema, inclusa la correlazione componente per componente con registrazione, prestazioni e guasti. Poiché il tracciamento semplifica l'identificazione visiva delle cause principali, viene dedicato meno tempo all'individuazione di tali cause. I team che hanno una visione dettagliata delle interazioni tra i componenti prendono decisioni migliori e più rapide durante la fase di risoluzione dei problemi. Le decisioni, ad esempio quando attivare il failover del ripristino di emergenza o dove implementare in modo più efficace le strategie di riparazione automatica, possono essere migliorate analizzando le tracce dei sistemi; ciò ottimizza in ultima analisi la soddisfazione dei clienti nei confronti dei servizi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I team che gestiscono le applicazioni distribuite possono utilizzare strumenti di tracciamento per definire un identificatore di correlazione, raccogliere le tracce delle richieste e creare mappe di servizio dei componenti connessi. Tutti i componenti dell'applicazione devono essere inclusi nelle tracce delle richieste, inclusi client di servizio, gateway middleware e router di eventi, componenti di elaborazione e archiviazione, tra cui gli archivi e i database dei valori chiave. Includi canary Synthetics o metriche RUM (Real-User Monitoring) nella configurazione del tracciamento end-to-end per misurare le interazioni e la latenza dei client remoti in modo da poter valutare con precisione le prestazioni dei tuoi sistemi rispetto agli accordi sul livello di servizio (SLA) e agli obiettivi corrispondenti.

Puoi utilizzare AWS X-Ray e i servizi di strumentazione di Monitoraggio delle applicazioni Amazon CloudWatch per avere una visione completa delle richieste man mano che vengono inviate all'applicazione. X-Ray raccoglie la telemetria delle applicazioni e consente di visualizzare e filtrare i dati corrispondenti tra payload, funzioni, tracce, servizi e API. L'acquisizione dei dati telemetrici può essere attivata per i componenti di sistema senza codice o a uso limitato di codice. Monitoraggio delle applicazioni CloudWatch include ServiceLens per integrare le tracce con metriche, log e allarmi. La funzionalità Monitoraggio delle applicazioni CloudWatch include anche elementi Synthetics per monitorare gli endpoint e le API, oltre alle metriche RUM (Real-User Monitoring) per instrumentare i client delle applicazioni Web.

Passaggi dell'implementazione

- Utilizza AWS X-Ray su tutti i servizi nativi supportati come Amazon S3, AWS Lambda e Amazon API Gateway. Questi servizi AWS consentono a X-Ray di attivare opzioni di configurazione utilizzando l'infrastruttura come codice, AWS SDK o la AWS Management Console.
- Esegui l'instrumentazione delle applicazioni AWS Distro per Open Telemetry e X-Ray o degli agenti di raccolta di terze parti.
- Consulta la Guida per gli sviluppatori AWS X-Ray per l'implementazione di linguaggi di programmazione specifici. Queste sezioni della documentazione descrivono come instrumentare le richieste HTTP, le query SQL e altri processi specifici del linguaggio di programmazione delle applicazioni.
- Usa il tracciamento X-Ray per i canary Synthetics di Amazon CloudWatch e le metriche RUM Amazon CloudWatch per analizzare il percorso delle richiesta dal client dell'utente finale attraverso l'infrastruttura AWS downstream.

- Configura le metriche CloudWatch e gli allarmi in base allo stato delle risorse e alla telemetria dei canary in modo che i team siano avvisati tempestivamente in merito ai problemi e possano, quindi, analizzare in dettaglio le tracce e le mappe dei servizi con ServiceLens.
- Abilita l'integrazione X-Ray per gli strumenti di tracciamento di terze parti come Datadog, New Relico Dynatrace se utilizzi strumenti di terze parti per la tua soluzione di tracciamento principale.

Risorse

Best practice correlate:

- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Documenti correlati:

- Che cos'è AWS X-Ray?
- Amazon CloudWatch: monitoraggio delle applicazioni
- Effettuare il debug con Amazon CloudWatch Synthetics e AWS X-Ray
- The Amazon Builders' Library: Dotazione dei sistemi distribuiti per la visibilità operativa
- Integrazione AWS X-Ray con altri servizi AWS
- AWS Distro per OpenTelemetry e AWS X-Ray
- Amazon CloudWatch: utilizzo del monitoraggio sintetico
- Amazon CloudWatch: utilizzo di CloudWatch RUM
- Installare i canary Amazon CloudWatch Synthetics e gli allarmi Amazon CloudWatch
- Oltre la disponibilità: comprendere e migliorare la resilienza dei sistemi distribuiti su AWS

Esempi correlati:

- One Observability Workshop

Video correlati:

- AWS re:Invent 2022 - How to monitor applications across multiple accounts (Come monitorare le applicazioni su più account)
- Come monitorare le tue applicazioni AWS

Strumenti correlati:

- AWS X-Ray
- Amazon CloudWatch
- Amazon Route 53

REL 7. Come si progetta il carico di lavoro per adattarsi ai cambiamenti della domanda?

Un carico di lavoro scalabile fornisce elasticità per aggiungere o rimuovere risorse automaticamente, in modo che vi sia una stretta corrispondenza con la domanda attuale in un dato momento.

Best practice

- REL07-BP01 Utilizzo dell'automazione per l'acquisizione o il dimensionamento delle risorse
- REL07-BP02 Ottenimento di risorse quando viene rilevata la compromissione di un carico di lavoro
- REL07-BP03 Ottenimento di risorse dopo aver rilevato che sono necessarie più risorse per un carico di lavoro
- REL07-BP04 Esecuzione di un test di carico sul carico di lavoro

REL07-BP01 Utilizzo dell'automazione per l'acquisizione o il dimensionamento delle risorse

Quando sostituisci risorse danneggiate o esegui il dimensionamento del carico di lavoro, puoi automatizzare il processo utilizzando servizi AWS gestiti, come Amazon S3 e AWS Auto Scaling. Puoi anche utilizzare strumenti di terze parti e SDK AWS per automatizzare il dimensionamento.

I servizi gestiti AWS includono Amazon S3, Amazon CloudFront, AWS Auto Scaling, AWS Lambda, Amazon DynamoDB, AWS Fargate e Amazon Route 53.

AWS Auto Scaling consente di rilevare e sostituire le istanze danneggiate. Inoltre, permette di creare piani di dimensionamento per le risorse, tra cui istanze e parchi istanze Amazon EC2 , attività Amazon ECS , tabelle e indici Amazon DynamoDB e repliche di Amazon Aurora .

Durante il dimensionamento di istanze EC2, assicurati di utilizzare più zone di disponibilità (preferibilmente almeno tre) e di aggiungere o rimuovere capacità per mantenere il bilanciamento tra queste zone. Anche le attività ECS o i pod Kubernetes (quando si utilizza Amazon Elastic Kubernetes Service) devono essere distribuiti su più zone di disponibilità.

Quando utilizzi AWS Lambda, le istanze subiscono un dimensionamento automatico. Ogni volta che viene ricevuta una notifica di evento per la funzione, AWS Lambda individua rapidamente la capacità libera all'interno del parco istanze di calcolo ed esegue il codice fino alla simultaneità allocata. Devi assicurarti che la simultaneità necessaria sia configurata sulla Lambda specifica e nelle tue Service Quotas.

Amazon S3 ricalibra automaticamente le risorse per gestire elevati tassi di richiesta. Ad esempio, l'applicazione può ottenere almeno 3.500 richieste PUT/COPY/POST/DELETE o 5.500 richieste GET /HEAD al secondo per prefisso in un bucket. Non ci sono limiti al numero di prefissi in un bucket. Puoi aumentare le prestazioni di lettura o scrittura parallelizzando le letture. Ad esempio, se crei 10 prefissi in un bucket Amazon S3 per parallelizzare le letture, potresti dimensionare le prestazioni di lettura a 55.000 richieste al secondo.

Configura e utilizza Amazon CloudFront o una rete di distribuzione di contenuti (CDN) attendibile. Una CDN può fornire tempi di risposta più rapidi agli utenti finali e può servire le richieste di contenuti dalla cache, riducendo così la necessità di dimensionare il carico di lavoro.

Anti-pattern comuni:

- Implementare gruppi Auto Scaling per la correzione automatica, ma senza elasticità.
- Utilizzare l'auto scaling per rispondere a grandi aumenti di traffico.
- Distribuire applicazioni altamente stateful, eliminando l'opzione di elasticità.

Vantaggi dell'adozione di questa best practice: L'automazione elimina il potenziale di errori manuali nella distribuzione e nella disattivazione delle risorse. L'automazione elimina il rischio di superamento dei costi e di rifiuto del servizio a causa della risposta lenta alle esigenze di distribuzione o disattivazione.

Livello di rischio associato se questa best practice non fosse adottata: Alta

Guida all'implementazione

- Configura e utilizza AWS Auto Scaling. In questo modo è possibile monitorare le applicazioni e regolare automaticamente la capacità per mantenere prestazioni stabili e prevedibili al minor costo possibile. Grazie ad AWS Auto Scaling, puoi configurare il dimensionamento delle applicazioni per più risorse in vari servizi.
  - Che cos'è AWS Auto Scaling?

    - Configura il dimensionamento automatico su serie di istanze Spot e istanze Amazon EC2, attività Amazon ECS, indici e tabelle Amazon DynamoDB, repliche Amazon Aurora e applicazioni Marketplace AWS, come applicabile.
        - Gestione automatica della capacità di throughput con DynamoDB Auto Scaling
            - Utilizza le operazioni delle API di servizi per specificare gli avvisi, le policy di ridimensionamento e i tempi di riscaldamento e raffreddamento.
- Utilizza Elastic Load Balancing. I sistemi di bilanciamento del carico possono distribuire il carico in base al percorso o alla connettività di rete.
    - Che cos'è Elastic Load Balancing?
        - Application Load Balancers può distribuire il carico per percorso.
            - What is an Application Load Balancer? (Che cos'è un Application Load Balancer?)
                - Configura un Application Load Balancer per distribuire il traffico su diversi carichi di lavoro in base a un percorso nello stesso nome di dominio.
                - Gli Application Load Balancers possono essere utilizzati per distribuire i carichi in modo da gestire la domanda attraverso l'integrazione con AWS Auto Scaling.
                    - Uso di un sistema di bilanciamento del carico con un gruppo Auto Scaling
        - I Network Load Balancer possono distribuire il carico in base alla connessione.
            - Che cos'è un Network Load Balancer?
                - Configura un Network Load Balancer per distribuire il traffico su diversi carichi di lavoro tramite TCP o per disporre di un set costante di indirizzi IP per il carico di lavoro.
                - I Network Load Balancer possono essere utilizzati per distribuire i carichi in modo da gestire la domanda attraverso l'integrazione con AWS Auto Scaling.
- Uso di un provider DNS altamente disponibile I nomi DNS consentono agli utenti di accedere ai carichi di lavoro utilizzando nomi anziché indirizzi IP e distribuire queste informazioni in un ambito definito, solitamente a livello globale per gli utenti del carico di lavoro.
    - Utilizza Amazon Route 53 o un provider DNS affidabile.
        - Che cos'è Amazon Route 53?
    - Utilizza Route 53 per gestire le distribuzioni CloudFront e i load balancer.
        - Individua i domini e i sottodomini da gestire.
        - Crea set di record appropriati utilizzando record ALIAS o CNAME.
            - Uso dei record

- Utilizza la rete globale AWS per ottimizzare il percorso dagli utenti alle applicazioni. AWS Global Accelerator monitora costantemente l'integrità degli endpoint delle applicazioni e reindirizza il traffico verso endpoint integri in meno di 30 secondi.
  - AWS Global Accelerator è un servizio che migliora la disponibilità e le prestazioni delle applicazioni con utenti locali o globali, fornendo indirizzi IP statici che fungono da punto di ingresso fisso agli endpoint delle applicazioni in una o più regioni Regioni AWS, ad esempio Application Load Balancers, Network Load Balancer o istanze Amazon EC2.
    - Che cos'è AWS Global Accelerator?
- Configura e utilizza Amazon CloudFront o una rete di distribuzione di contenuti (CDN) attendibile. Una rete di distribuzione di contenuti (CDN) può fornire tempi di risposta più rapidi agli utenti finali e soddisfare richieste di contenuti che possono causare un dimensionamento non necessario dei carichi di lavoro.
  - Che cos'è Amazon CloudFront?
    - Configura le distribuzioni di Amazon CloudFront per i carichi di lavoro oppure utilizza una CDN di terze parti.
      - Puoi limitare l'accesso ai tuoi carichi di lavoro in modo che siano accessibili solo da CloudFront utilizzando gli intervalli di indirizzi IP per CloudFront nelle policy di accesso o nei gruppi di sicurezza degli endpoint.

Risorse

Documenti correlati:

- Partner APN: partner per la creazione di soluzioni di elaborazione automatizzate
- AWS Auto Scaling: come funzionano i piani di dimensionamento
- Marketplace AWS: prodotti che possono essere utilizzati con Auto Scaling
- Gestione automatica della capacità di throughput con DynamoDB Auto Scaling
- Uso di un sistema di bilanciamento del carico con un gruppo Auto Scaling
- Che cos'è AWS Global Accelerator?
- Che cos'è Amazon EC2 Auto Scaling?
- Che cos'è AWS Auto Scaling?
- Che cos'è Amazon CloudFront?
- Che cos'è Amazon Route 53?
- Che cos'è Elastic Load Balancing?

- Che cos'è un Network Load Balancer?
- What is an Application Load Balancer? (Che cos'è un Application Load Balancer?)
- Uso dei record

REL07-BP02 Ottenimento di risorse quando viene rilevata la compromissione di un carico di lavoro

All'occorrenza, ridimensiona le risorse in modo reattivo se la disponibilità è influenzata per ripristinare la disponibilità del carico di lavoro.

Devi prima configurare i controlli dello stato e i criteri su questi controlli per indicare quando la disponibilità è influenzata dalla mancanza di risorse. Quindi invita il personale appropriato a dimensionare manualmente la risorsa o attivare l'automazione per dimensionarla automaticamente.

Il dimensionamento può essere regolato manualmente in base al carico di lavoro, ad esempio modificando il numero di istanze EC2 in un gruppo Auto Scaling o modificando la velocità di trasmissione effettiva di una tabella DynamoDB tramite la AWS Management Console o la AWS CLI. Tuttavia, l'automazione deve essere utilizzata ogni volta che è possibile (consulta Utilizzo dell'automazione per l'acquisizione o il dimensionamento delle risorse).

Risultato desiderato: le attività di dimensionamento (automatico o manuale) vengono avviate per ripristinare la disponibilità al rilevamento di un guasto o di un'esperienza degradata del cliente.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Implementa l'osservabilità e il monitoraggio su tutti i componenti del carico di lavoro, per monitorare l'esperienza del cliente e rilevare i guasti. Definisci le procedure (manuali o automatiche) che dimensionano le risorse richieste. Per ulteriori informazioni, consulta REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti.

Passaggi dell'implementazione

- Definisci le procedure (manuali o automatiche) che dimensionano le risorse richieste.
  - Le procedure di dimensionamento dipendono da come sono progettati i diversi componenti del carico di lavoro.
  - Le procedure di dimensionamento variano anche a seconda della tecnologia sottostante utilizzata.

- I componenti che utilizzano AWS Auto Scaling possono impiegare piani di dimensionamento per configurare una serie di istruzioni per dimensionare le risorse. Se utilizzi AWS CloudFormation o aggiungi tag alle risorse AWS, puoi impostare piani di dimensionamento per diversi set di risorse, per ogni applicazione. Auto Scaling fornisce raccomandazioni per strategie di dimensionamento personalizzate per ogni risorsa. Dopo aver creato il piano, Auto Scaling combina i metodi di dimensionamento dinamico e predittivo per supportare la tua strategia di dimensionamento. Per maggiori dettagli, consulta Come funzionano i piani di dimensionamento.
- Amazon EC2 Auto Scaling garantisce che sia disponibile il numero corretto di istanze Amazon EC2 per gestire il carico dell'applicazione. È possibile creare raccolte di istanze EC2, denominate gruppi Auto Scaling. Puoi specificare il numero minimo e massimo di istanze in ogni gruppo Auto Scaling; Amazon EC2 Auto Scaling garantisce che il gruppo non superi mai questi limiti. Per maggiori dettagli, vedi What is Amazon EC2 Auto Scaling?
- Il dimensionamento automatico Amazon DynamoDB utilizza il servizio Application Auto Scaling per regolare dinamicamente la capacità effettiva di trasmissione allocata per tuo conto, in risposta ai modelli di traffico effettivi. Ciò consente a una tabella o a un indice secondario globale di aumentare la capacità di lettura e scrittura assegnata per gestire aumenti di traffico improvvisi, senza limitazione della larghezza di banda della rete. Per maggiori dettagli, consulta Gestione automatica della capacità effettiva di trasmissione con il dimensionamento automatico di DynamoDB.

Risorse

Best practice correlate:

- REL07-BP01 Utilizzo dell'automazione per l'acquisizione o il dimensionamento delle risorse
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Documenti correlati:

- AWS Auto Scaling: Come funzionano i piani di dimensionamento
- Gestione automatica della capacità effettiva di trasmissione con il dimensionamento automatico di DynamoDB
- What is Amazon EC2 Auto Scaling?

REL07-BP03 Ottenimento di risorse dopo aver rilevato che sono necessarie più risorse per un carico di lavoro

Dimensiona le risorse in modo proattivo per soddisfare la domanda ed evitare l'impatto sulla disponibilità.

Molti servizi AWS dimensionano automaticamente le risorse per soddisfare la domanda. Se si utilizzano istanze Amazon EC2 o cluster Amazon ECS, puoi configurare la scalabilità automatica di tali istanze in base ai parametri di utilizzo corrispondenti alla richiesta del carico di lavoro. Per Amazon EC2, è possibile impiegare l'utilizzo medio della CPU, il conteggio delle richieste del sistema di bilanciamento del carico o la larghezza di banda di rete per aumentare (o ridurre) le istanze EC2. Per Amazon ECS, è possibile impiegare l'utilizzo medio della CPU, il conteggio delle richieste del load balancer e l'utilizzo della memoria per aumentare orizzontalmente (o ridurre orizzontalmente) le attività ECS. Utilizzando il dimensionamento automatico di destinazione su AWS, l'autoscaler si comporta come un termostato domestico, aggiungendo o rimuovendo risorse per mantenere il valore di destinazione (ad esempio, il 70% di utilizzo della CPU) specificato.

AWS Auto Scaling può anche eseguire l' Auto Scaling predittivo, che utilizza il machine learning per analizzare il carico di lavoro cronologico di ciascuna risorsa e prevede regolarmente il carico futuro per i due giorni successivi.

La legge di Little aiuta a calcolare il numero di istanze di calcolo (istanze EC2, funzioni Lambda simultanee, ecc.) necessarie.

$L = \lambda W$

L = numero di istanze (o simultaneità media nel sistema)

λ = velocità media alla quale arrivano le richieste (richieste/sec)

W = tempo medio trascorso da ogni richiesta nel sistema (sec)

Ad esempio, a 100 rps, se ogni richiesta impiega 0,5 secondi per l'elaborazione, avrai bisogno di 50 istanze per tenere il passo con la domanda.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

- Ottieni risorse dopo aver rilevato che sono necessarie più risorse per un carico di lavoro Dimensiona le risorse in modo proattivo per soddisfare la domanda ed evitare l'impatto sulla disponibilità.

- Valuta quante risorse di calcolo sono necessarie (simultaneità di calcolo) per gestire un determinato tasso di richiesta
  - Telling Stories About Little's Law
- Quando disponi di un modello cronologico per l'utilizzo, imposta il dimensionamento programmato per il dimensionamento automatico Amazon EC2.
  - Dimensionamento programmato per Amazon EC2 Auto Scaling
- Utilizza il dimensionamento predittivo di AWS.
  - Dimensionamento predittivo per EC2, alimentato dal machine learning

Risorse

Documenti correlati:

- AWS Auto Scaling: come funzionano i piani di dimensionamento
- Marketplace AWS: prodotti che possono essere utilizzati con Auto Scaling
- Gestione automatica della capacità di throughput con DynamoDB Auto Scaling
- Dimensionamento predittivo per EC2, alimentato dal machine learning
- Dimensionamento programmato per Amazon EC2 Auto Scaling
- Telling Stories About Little's Law
- Che cos'è Amazon EC2 Auto Scaling?

REL07-BP04 Esecuzione di un test di carico sul carico di lavoro

Adotta un metodo di test del carico per misurare se l'attività di dimensionamento soddisfa i requisiti del carico di lavoro.

È importante eseguire test di carico prolungati. I test di carico devono rilevare il punto di rottura e testare le prestazioni del carico di lavoro. AWS consente di creare facilmente ambienti di test temporanei che riproducono la scala del carico di lavoro di produzione. Nel cloud, puoi creare un ambiente di test su scala produttiva on demand, completare i test e disattivare le risorse. Poiché paghi per l'ambiente di test solo quando è in esecuzione, puoi simulare un ambiente live a un costo notevolmente inferiore rispetti ai test in locale.

I test di carico in produzione dovrebbero anche essere considerati come parte dei game day in cui il sistema di produzione viene messo alla prova, durante le ore di utilizzo inferiore del cliente, con tutto il personale a disposizione per interpretare i risultati e risolvere eventuali problemi che si presentano.

Anti-pattern comuni:

- Eseguire test di carico su distribuzioni che non presentano la stessa configurazione della tua produzione.
- Eseguire test di carico solo su singole parti del carico di lavoro e non sulla sua interezza.
- Eseguire test di carico con un sottoinsieme di richieste e non con un set rappresentativo delle richieste effettive.
- Eseguire test di carico su un fattore di sicurezza di poco superiore al carico previsto.

Vantaggi dell'adozione di questa best practice: Saprai quali sono i componenti dell'architettura che non funzionano sotto carico e potrai identificare per tempo i parametri che indicano l'avvicinamento al carico in questione, così da affrontare il problema e prevenire l'impatto dell'esito negativo.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Esegui test di carico per identificare quali aspetti del carico di lavoro indicano la necessità di aggiungere o rimuovere capacità. Il test di carico deve avere un traffico rappresentativo simile a quello che ricevi nella produzione. Aumenta il carico mentre osservi i parametri implementati per stabilire quale di questi indica quando è necessario aggiungere o rimuovere risorse.
  - Distributed Load Testing on AWS (Test di carico distribuito su AWS): simula migliaia di utenti connessi
    - Identifica la combinazione di richieste. Potresti avere diverse combinazioni di richieste, quindi dovresti esaminare vari intervalli di tempo per identificare la combinazione di traffico.
    - Implementa un driver di caricamento. Puoi utilizzare codice personalizzato, software open source o software commerciale per implementare un driver di carico.
    - Esegui un test di carico iniziale con una capacità ridotta. Puoi vedere alcuni effetti immediati applicando il carico su una capacità inferiore, possibilmente pari a un'istanza o a un container.
    - Esegui un test di carico con una capacità maggiore. Gli effetti saranno diversi su un carico distribuito, quindi è necessario eseguire il test in condizioni quanto più simili possibili all'ambiente del prodotto.

Risorse

Documenti correlati:

- Distributed Load Testing on AWS (Test di carico distribuito su AWS): simula migliaia di utenti connessi
- Load testing applications

Video correlati:

- AWS Summit ANZ 2023: Accelerate with confidence through AWS Distributed Load Testing

REL 8. In che modo implementare le modifiche?

Per implementare nuove funzionalità e verificare che i carichi di lavoro e l'ambiente operativo eseguano software noti e che sia possibile applicare patch o sostituirli in modo prevedibile, sono necessarie modifiche controllate. Se invece non sono controllate, risulta difficile prevederne l'effetto o risolvere eventuali problemi che causano.

Best practice

- REL08-BP01 Utilizzo di runbook per attività standard come l'implementazione
- REL08-BP02 Esecuzione di test funzionali come parte integrante dell'implementazione
- REL08-BP03 Esecuzione di test di resilienza come parte integrante dell'implementazione
- REL08-BP04 Esecuzione dell'implementazione utilizzando un'infrastruttura immutabile
- REL08-BP05 Implementazione delle modifiche tramite automazione

REL08-BP01 Utilizzo di runbook per attività standard come l'implementazione

I runbook sono le procedure predefinite per ottenere risultati specifici. Utilizza i runbook per eseguire attività standard, o manualmente o automaticamente. Alcuni esempi includono l'implementazione di un carico di lavoro, l'applicazione di patch a un carico di lavoro o la realizzazione di modifiche DNS.

Ad esempio, metti in atto processi per garantire la sicurezza del rollback durante le distribuzioni. Garantire la possibilità di eseguire il rollback di una distribuzione senza interruzioni per i clienti è fondamentale per rendere un servizio affidabile.

Per le procedure di runbook, inizia da un processo manuale valido ed efficace, implementalo nel codice e attivalo per l'esecuzione automatica, se necessario.

Anche per carichi di lavoro sofisticati e altamente automatizzati, i runbook rimangono utili per eseguire game day o soddisfare rigorosi requisiti di reportistica e audit.

Tieni presente che i playbook vengono utilizzati in risposta a incidenti specifici e i runbook vengono utilizzati per ottenere risultati specifici. Spesso, i runbook sono per attività di routine, mentre i playbook vengono utilizzati per rispondere a eventi non di routine.

Anti-pattern comuni:

- Eseguire modifiche impreviste alla configurazione nella produzione.
- Ignorare le fasi del piano per velocizzare l'implementazione, compromettendone la riuscita.
- Apportare modifiche senza testarne l'annullamento.

Vantaggi dell'adozione di questa best practice: Una pianificazione efficace aumenta la capacità di eseguire correttamente la modifica, perché sei a conoscenza di tutti i sistemi interessati. Convalidare la modifica negli ambienti di test aumenta la sicurezza.

Livello di rischio associato se questa best practice non fosse adottata: Alta

Guida all'implementazione

- Abilita risposte coerenti e tempestive agli eventi noti documentando le procedure nei runbook.
  - Framework AWS Well-Architected – Concetti – Runbook
- Uso del principio di infrastruttura come codice per definire l'infrastruttura Utilizzando AWS CloudFormation o una terza parte affidabile per definire la tua infrastruttura, puoi utilizzare un software per il controllo delle versioni per gestire le versioni e tenere traccia delle modifiche.
  - Utilizza AWS CloudFormation o un provider di terze parti affidabile per definire l'infrastruttura.
    - Che cos'è AWS CloudFormation?
  - Crea modelli unici e disaccoppiati, utilizzando solidi principi di progettazione del software.
    - Stabilisci le autorizzazioni, i modelli e le parti responsabili dell'implementazione
      - Controllo degli accessi con AWS Identity and Access Management
    - Utilizza un controllo sorgente come AWS CodeCommit o uno strumento di terze parti affidabili per il controllo delle versioni.
      - Che cos'è AWS CodeCommit?

Risorse

Documenti correlati:

- Partner APN: partner per la creazione di soluzioni di distribuzione automatizzate
- Marketplace AWS: prodotti per l'automazione delle distribuzioni
- Framework AWS Well-Architected – Concetti – Runbook
- Che cos'è AWS CloudFormation?
- Che cos'è AWS CodeCommit?

Esempi correlati:

- Automating operations with Playbooks and Runbooks (Automazione delle operazioni con Playbook e Runbook)

REL08-BP02 Esecuzione di test funzionali come parte integrante dell'implementazione

I test funzionali vengono eseguiti come parte integrante dell'implementazione automatizzata. Se non vengono soddisfatti i criteri di esito positivo, la pipeline viene arrestata o ripresa dall'inizio. Questi test vengono eseguiti in un ambiente di pre-produzione, gestito per fasi prima della produzione nella pipeline. Idealmente, questa operazione viene eseguita come parte di una pipeline di implementazione.

Risultato desiderato: utilizzi l'automazione per eseguire test funzionali e i dati dei test associati riducono la durata e il costo dei test e migliorano l'accuratezza dei risultati. Integri i test funzionali come parte del processo di implementazione per automatizzare le pipeline di rilascio per l'esecuzione di aggiornamenti rapidi e affidabili di applicazioni e infrastrutture.

Anti-pattern comuni:

- I test vengono eseguiti manualmente al di fuori della pipeline di implementazione.
- Non esegui le fasi di test nell'automazione tramite flussi di lavoro manuali di emergenza.
- Non segui i piani e i processi di test stabiliti a favore di tempistiche accelerate.

Vantaggi dell'adozione di questa best practice: i test funzionali verificano che il sistema funzioni secondo i requisiti specificati. Vengono utilizzati per verificare in modo coerente il funzionamento previsto di componenti quali interfacce utente, API, database e codice sorgente. Quando esamini questi componenti del sistema, i test funzionali verificano che ciascuna funzionalità si comporti come previsto, tutelando sia le esigenze degli utenti sia l'integrità del software. Integra i test funzionali come parte dell'implementazione regolare e utilizza l'automazione per implementare tutte le modifiche, riducendo i potenziali errori umani.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Esegui test funzionali come parte integrante dell'implementazione. I test funzionali vengono eseguiti come parte integrante dell'implementazione automatizzata. Se non vengono soddisfatti i criteri di esito positivo, la pipeline viene arrestata o ripristinata. AWS CodePipeline fornisce una pipeline di distribuzione continua per i test automatizzati, che consente ai tester di automatizzare l'intero processo di test e implementazione. Si integra con i servizi AWS come AWS CodeBuild e AWS CodeDeploy per automatizzare le fasi di creazione, test e implementazione del ciclo di vita di sviluppo del software.

Passaggi dell'implementazione

- Configura la pipeline: configura le fasi di origine, creazione, test e implementazione utilizzando la console AWS CodePipeline o AWS Command Line Interface (CLI).
  - Definisci l'origine: con AWS CodePipeline puoi recuperare automaticamente il codice sorgente da sistemi di controllo delle versioni come GitHub, AWS CodeCommit o Bitbucket e assicurarti che per i test venga sempre utilizzato il codice più recente.
  - Automatizza build e test: AWS CodeBuild può creare e testare automaticamente il codice e generare i report di test. Supporta i framework di test più diffusi come JUnit, NUnit e TestNG.
  - Implementa il codice: una volta che il codice è stato creato e testato, AWS CodeDeploy può implementarlo nell'ambiente di test, incluse le istanze Amazon EC2, le funzioni AWS Lambda o i server on-premises.
  - Monitora le pipeline: con AWS CodePipeline puoi monitorare l'avanzamento della pipeline e lo stato di ogni fase. Puoi anche utilizzare i controlli di qualità per bloccare la pipeline in base allo stato di esecuzione dei test. Puoi anche ricevere notifiche per qualsiasi errore che si verifica durante l'esecuzione o il completamento della pipeline.

Risorse

Documenti correlati:

- Use AWS CodePipeline with AWS CodeBuild to test code and run builds
- Logging and monitoring in AWS CodeBuild
- Indicators for functional testing

REL08-BP03 Esecuzione di test di resilienza come parte integrante dell'implementazione

Integra i test di resilienza introducendo consapevolmente errori nel sistema per misurarne la capacità in caso di scenari destabilizzanti. I test di resilienza, diversamente dai test funzionali e dagli unit test che di solito sono integrati nei cicli di implementazione, si concentrano sull'identificazione di errori imprevisti nel sistema. Puoi iniziare l'integrazione dei test di resilienza nella fase di pre-produzione, ma stabilisci l'obiettivo di implementare questi test in produzione durante le giornate di gioco.

Risultato desiderato: i test di resilienza favoriscono la fiducia nella capacità del sistema di reggere al danneggiamento in produzione. Gli esperimenti identificano i punti di debolezza che potrebbero causare errori, consentendoti di migliorare il sistema per mitigare automaticamente ed efficacemente errori e danneggiamento.

Anti-pattern comuni:

- Mancanza di osservabilità e monitoraggio nei processi di implementazione.
- Dipendenza dagli esseri umani per risolvere gli errori del sistema.
- Meccanismi di analisi di scarsa qualità.
- Supporto per i problemi noti del sistema e mancanza di sperimentazione per identificare eventuali incognite.
- Identificazione degli errori, ma nessuna risoluzione.
- Nessuna documentazione degli esiti e dei runbook.

Vantaggi dell'adozione di questa best practice: i test di resilienza integrati nelle implementazioni aiutano a identificare problemi non noti del sistema che altrimenti passano inosservati, causando tempi di inattività in produzione. L'identificazione di queste incognite nel sistema ti consente di documentare gli esiti, integrare i test nel processo CI/CD e creare runbook che semplificano la mitigazione attraverso meccanismi efficienti e ripetibili.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I moduli di test di resilienza più comuni che possono essere integrati nelle implementazioni del sistema sono il ripristino di emergenza e l'ingegneria del caos.

- Includi gli aggiornamenti ai piani di ripristino di emergenza e alle procedure operative standard (SOP) con qualsiasi implementazione significativa.

- Integra i test di affidabilità nelle pipeline di implementazione automatizzate. Servizi come AWS Resilience Hub possono essere integrati nella pipeline CI/CD per stabilire valutazioni continue della resilienza che vengono eseguite automaticamente come parte di ogni implementazione.
- Definisci le applicazioni in AWS Resilience Hub. Le valutazioni della resilienza generano frammenti di codice che consentono di creare procedure di ripristino come i documenti AWS Systems Manager per le applicazioni e forniscono un elenco di controlli e allarmi Amazon CloudWatch consigliati.
- Una volta aggiornati i piani di ripristino di emergenza e le SOP, completa i test di ripristino di emergenza per verificarne l'efficacia. I test di ripristino di emergenza consentono di determinare se è possibile ripristinare il sistema dopo un evento e tornare alle normali operazioni. Puoi simulare varie strategie di ripristino di emergenza e determinare se la pianificazione è sufficiente a soddisfare i requisiti di operatività. Le strategie di ripristino di emergenza più comuni includono backup e ripristino, pilot light, cold standby, warm standby, standby a caldo e attivo-attivo e si differenziano tutte per costi e complessità. Prima dei test di ripristino di emergenza, ti consigliamo di definire l'obiettivo del tempo di ripristino (RTO) e l'obiettivo del punto di ripristino (RPO) per semplificare la scelta della strategia da simulare. AWS offre strumenti di ripristino di emergenza come AWS Elastic Disaster Recovery per aiutarti a iniziare la pianificazione e i test.
- Gli esperimenti di ingegneria del caos introducono interruzioni nel sistema, come interruzioni di rete ed errori del servizio. Simulando con gli errori controllati, puoi scoprire le vulnerabilità del sistema contenendo al contempo l'impatto degli errori inseriti. Proprio come le altre strategie, esegui le simulazioni controllate degli errori in ambienti non di produzione utilizzando servizi come AWS Fault Injection Service per acquisire sicurezza prima dell'implementazione in produzione.

Risorse

Documenti correlati:

- Experiment with failure using resilience testing to build recovery preparedness
- Continually assessing application resilience with AWS Resilience Hub and AWS CodePipeline
- Disaster recovery (DR) architecture on AWS, part 1: Strategies for recovery in the cloud
- Verify the resilience of your workloads using Chaos Engineering
- Principles of Chaos Engineering
- Chaos Engineering Workshop

Video correlati:

- AWS re:Invent 2020: Testing Resilience using Chaos Engineering
- Improve Application Resilience with AWS Fault Injection Service
- Prepare & Protect Your Applications From Disruption With AWS Resilience Hub

REL08-BP04 Esecuzione dell'implementazione utilizzando un'infrastruttura immutabile

L'infrastruttura immutabile è un modello che richiede che non vengano applicati aggiornamenti, patch di sicurezza o modifiche di configurazione sui carichi di lavoro di produzione. Quando è necessaria una modifica, l'architettura viene costruita su una nuova infrastruttura e distribuita alla produzione.

Segui una strategia di implementazione dell'infrastruttura immutabile per aumentare l'affidabilità, la coerenza e la riproducibilità nelle implementazioni dei carichi di lavoro.

Risultato desiderato: con un'infrastruttura immutabile, non sono consentite modifiche locali (in-place) per l'esecuzione delle risorse dell'infrastruttura all'interno di un carico di lavoro. Invece, quando è necessaria una modifica, un nuovo set di risorse infrastrutturali aggiornate contenente tutte le modifiche necessarie viene implementato in parallelo alle risorse esistenti. Questa implementazione viene convalidata automaticamente e, in caso di successo, il traffico viene gradualmente trasferito al nuovo set di risorse.

Questa strategia di implementazione si applica, ad esempio, agli aggiornamenti software, alle patch di sicurezza, alle modifiche apportate all'infrastruttura, agli aggiornamenti della configurazione e agli aggiornamenti delle applicazioni.

Anti-pattern comuni:

- Implementazione locale (in-place) di modifiche alle risorse dell'infrastruttura in esecuzione.

Vantaggi dell'adozione di questa best practice:

- Maggiore coerenza tra ambienti: poiché non vi sono differenze nelle risorse dell'infrastruttura tra ambienti, la coerenza aumenta e i test risultano semplificati.
- Riduzione delle deviazioni di configurazione: sostituendo le risorse dell'infrastruttura con una configurazione nota e controllata dalla versione, l'infrastruttura viene reimpostata a uno stato noto, testato e attendibile, evitando in questo modo deviazioni di configurazione.
- Implementazioni atomiche affidabili: le implementazioni vengono completate correttamente o, in caso contrario, non generano alcun cambiamento, aumentando così la coerenza e l'affidabilità nel processo di implementazione.

- Implementazioni semplificate: le implementazioni sono semplificate perché non devono supportare gli aggiornamenti. Gli aggiornamenti sono solo nuove distribuzioni.
- Implementazioni più sicure con processi di rollback e ripristino rapidi: le implementazioni sono più sicure perché la versione funzionante precedente non viene modificata. Puoi eseguire il rollback se vengono rilevati errori.
- Potenziamento del profilo di sicurezza: non consentendo modifiche all'infrastruttura, i meccanismi di accesso remoto (come SSH) possono essere disabilitati. Questo riduce il vettore di attacco, migliorando il profilo di sicurezza dell'organizzazione.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Automazione

Quando si definisce una strategia di implementazione immutabile dell'infrastruttura, si consiglia di utilizzare l'automazione il più possibile per aumentare la riproducibilità e ridurre al minimo i potenziali errori umani. Per maggiori dettagli, consulta REL08-BP05 Implementazione delle modifiche tramite automazione e Automazione di implementazioni pratiche e sicure.

Con il modello Infrastructure as code (IaC), le fasi di provisioning, orchestrazione e implementazione dell'infrastruttura sono definite in modo programmatico, descrittivo e dichiarativo e archiviate in un sistema di controllo del codice sorgente. L'utilizzo del modello Infrastructure as code (IaC) semplifica l'automazione dell'implementazione dell'infrastruttura e aiuta a raggiungere l'immutabilità dell'infrastruttura.

Schemi di implementazione

Quando è richiesta una modifica del carico di lavoro, la strategia di implementazione immutabile dell'infrastruttura impone l'implementazione di un nuovo set di risorse dell'infrastruttura, comprese tutte le modifiche necessarie. È importante che questo nuovo set di risorse si basi su un modello di implementazione che riduca al minimo l'impatto sugli utenti. Esistono due strategie principali per questa implementazione:

Implementazione Canary: pratica di indirizzare un piccolo numero di clienti alla nuova versione, in genere in esecuzione su una singola istanza di servizio (la release Canary). Quindi analizzerai in modo approfondito le modifiche di comportamento o gli errori generati. Puoi rimuovere il traffico dalla release Canary in caso di problemi critici e reindirizzare gli utenti alla versione precedente. Se l'implementazione viene completata correttamente, puoi continuare l'implementazione

alla velocità desiderata, monitorando le modifiche alla ricerca di errori, fino al completamento dell'implementazione. AWS CodeDeploy può essere configurato con una configurazione di implementazione che abilita un'implementazione Canary.

Implementazione blu/verde: simile all'implementazione Canary, tranne per il fatto che viene implementato in parallelo un intero parco istanze dell'applicazione. Puoi alternare le distribuzioni tra i due stack (blue e green). Ancora una volta, puoi inviare il traffico alla nuova versione e tornare alla versione precedente in caso di problemi con la distribuzione. Generalmente, tutto il traffico viene trasferito contemporaneamente, tuttavia puoi anche utilizzare frazioni del traffico verso ciascuna versione per comporre l'adozione della nuova versione utilizzando le funzionalità di indirizzamento DNS ponderato di Amazon Route 53. AWS CodeDeploy e AWS Elastic Beanstalk possono essere configurati con una configurazione di distribuzione che abilita un'implementazione blu/verde.

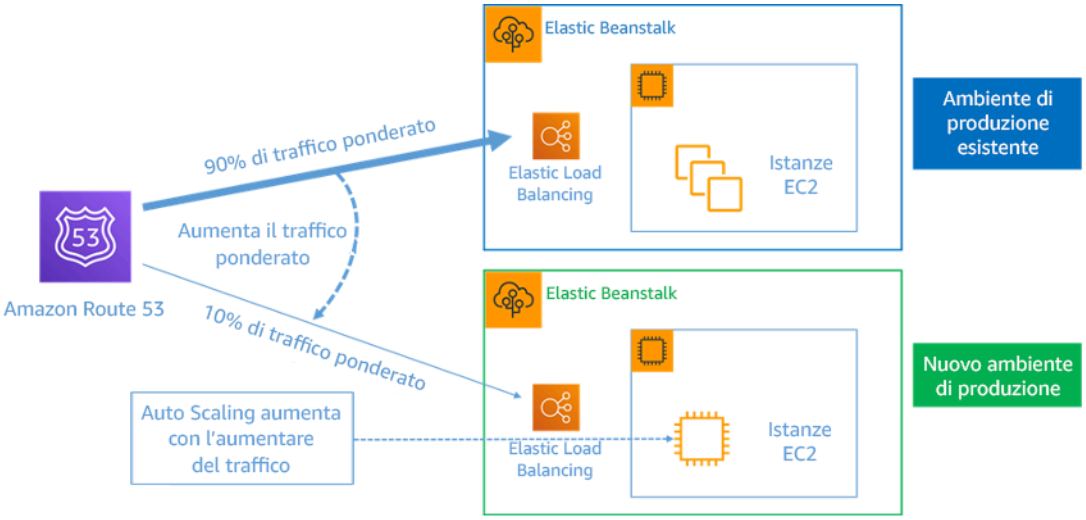


Figura 8. Implementazione blu/verde con AWS Elastic Beanstalk e Amazon Route 53

Rilevamento della deviazione

La deviazione è definita come qualsiasi modifica che fa sì che una risorsa dell'infrastruttura abbia uno stato o una configurazione diversi dal previsto. Qualsiasi tipo di modifica non gestita della configurazione è contraria al concetto di infrastruttura immutabile e tale modifica dovrebbe essere individuata e corretta per implementare con successo l'infrastruttura immutabile.

Passaggi dell'implementazione

- Non autorizzare la modifica locale (in-place) delle risorse dell'infrastruttura in esecuzione.
  - Puoi usare AWS Identity and Access Management (IAM) per specificare chi o cosa può accedere a servizi e risorse in AWS, gestire centralmente le autorizzazioni a elevata granularità e analizzare l'accesso per perfezionare le autorizzazioni in AWS.

- Automatizza l'implementazione delle risorse dell'infrastruttura per aumentare la riproducibilità e ridurre al minimo i potenziali errori umani.
  - Come descritto nel whitepaper Introduzione a DevOps in AWS, l'automazione è un elemento fondamentale per i servizi AWS ed è supportata internamente in tutti i servizi, le funzionalità e le offerte.
  - La preparazione preliminare delle Amazon Machine Image (AMI) può velocizzare i tempi di avvio. EC2 Image Builder è un servizio AWS completamente gestito che consente di automatizzare la creazione, la manutenzione, la convalida, la condivisione e l'implementazione di AMI personalizzate, sicure e aggiornate per Linux o Windows.
  - Alcuni dei servizi che supportano l'automazione sono:
    - AWS Elastic Beanstalk è un servizio per implementare e dimensionare rapidamente applicazioni Web sviluppate con Java, .NET, PHP, Node.js, Python, Ruby, Go e Docker su server tradizionali come Apache, NGINX, Passenger e IIS.
    - AWS Proton aiuta i team della piattaforma a connettere e coordinare tutti i diversi strumenti di cui i team di sviluppo hanno bisogno per il provisioning dell'infrastruttura, le implementazioni del codice, il monitoraggio e gli aggiornamenti. AWS Proton abilita il provisioning e l'implementazione basati sul modello IaC di applicazioni serverless e basate su container.
  - L'utilizzo del modello Infrastructure as code (IaC) semplifica l'automazione dell'implementazione dell'infrastruttura e aiuta a raggiungere l'immutabilità dell'infrastruttura. AWS fornisce servizi che consentono la creazione, l'implementazione e la manutenzione dell'infrastruttura in modo programmatico, descrittivo e dichiarativo.
    - AWS CloudFormation aiuta gli sviluppatori a creare risorse AWS in modo ordinato e prevedibile. Le risorse sono scritte in file di testo utilizzando il formato JSON o YAML. I modelli richiedono una sintassi e una struttura specifiche che dipendono dai tipi di risorse create e gestite. Crei le tue risorse in formato JSON o YAML con qualsiasi editor di codice, ad esempio AWS Cloud9, e le inserisci in un sistema di controllo delle versioni. A questo punto, CloudFormation crea i servizi specificati in modo sicuro e ripetibile.
    - AWS Serverless Application Model (AWS SAM) è un framework open source che puoi utilizzare per creare applicazioni serverless in AWS. AWS SAM si integra con altri servizi AWS ed è un'estensione di AWS CloudFormation.
    - AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo di software open source per modellare ed effettuare il provisioning delle risorse delle applicazioni cloud utilizzando linguaggi di programmazione noti. È possibile utilizzare AWS CDK per modellare l'infrastruttura dell'applicazione mediante TypeScript, Python, Java e.NET. AWS CDK utilizza AWS CloudFormation in background per fornire risorse in modo sicuro e ripetibile.

    - AWS Cloud Control API introduce un set comune di API Create, Read, Update, Delete and List (CRUDL) per aiutare gli sviluppatori a gestire la propria infrastruttura cloud in modo semplice e coerente. Le API Cloud Control API comuni consentono agli sviluppatori di gestire in modo uniforme il ciclo di vita di AWS e i servizi di terze parti.
- Applica modelli di implementazione che riducano al minimo l'impatto sugli utenti.
    - Implementazioni canary:
        - Configura un'implementazione di una release canary API Gateway
        - Crea una pipeline con implementazioni canary per Amazon ECS mediante AWS App Mesh
    - Implementazioni blu/verdi: il whitepaper relativo alle implementazioni blu/verdi in AWSdescrive esempi di tecniche per applicare le strategie di implementazione blu/verde.
- Rileva le deviazioni a livello di configurazione o stato. Per maggiori dettagli, consulta Rilevamento di modifiche non gestite della configurazione di stack e risorse.

Risorse

Best practice correlate:

- REL08-BP05 Implementazione delle modifiche tramite automazione

Documenti correlati:

- Automazione di implementazioni pratiche e sicure
- Utilizzo di AWS CloudFormation per creare un'infrastruttura immutabile presso Nubank
- Scrittura dell’infrastruttura come codice
- Implementazione di un allarme per rilevare automaticamente la deviazione negli stack AWS CloudFormation

Video correlati:

- AWS re:Invent 2020: Reliability, consistency, and confidence through immutability

REL08-BP05 Implementazione delle modifiche tramite automazione

Le distribuzioni e l'applicazione di patch sono automatizzate per eliminare l'impatto negativo.

Apportare modifiche ai sistemi produttivi è una delle maggiori aree di rischio per molte organizzazioni. Riteniamo che le implementazioni siano un problema prioritario da risolvere insieme ai problemi aziendali affrontati dal software. Oggi, ciò significa l'uso dell'automazione ovunque sia pratica nelle operazioni, inclusi test e implementazione di modifiche, aggiunta o rimozione di capacità e migrazione dei dati.

Risultato desiderato: integri la sicurezza dell'implementazione automatizzata nel processo di rilascio con test approfonditi di pre-produzione, rollback automatici e implementazioni di produzione scaglionate. Questa automazione riduce al minimo il potenziale impatto sulla produzione causato da implementazioni non riuscite e gli sviluppatori non devono più monitorare attivamente le implementazioni in produzione.

Anti-pattern comuni:

- Esegui le modifiche manualmente.
- Non esegui le fasi nell'automazione tramite flussi di lavoro manuali di emergenza.
- Non segui i piani e i processi stabiliti a favore di tempistiche accelerate.
- Esegui implementazioni successive rapide senza attendere i tempi di incorporamento.

Vantaggi dell'adozione di questa best practice: quando utilizzi l'automazione per implementare tutte le modifiche, elimini il rischio di errori umani e fornisci la possibilità di eseguire i test prima di modificare la produzione. L'esecuzione di questo processo prima del passaggio in produzione verifica che i piani siano completi. Inoltre, il rollback automatico del processo di rilascio può identificare i problemi di produzione e riportare il carico di lavoro allo stato operativo precedente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Automatizzazione della pipeline di distribuzione Le pipeline di implementazione permettono di richiamare test automatici, rilevare le anomalie e interrompere la pipeline a una determinata fase prima dell'implementazione in produzione o eseguire automaticamente il ripristino di una modifica. Parte integrante è l'adozione della cultura basata su integrazione continua e distribuzione/implementazione continua (CI/CD), in cui un commit o una modifica del codice passa attraverso vari controlli automatizzati dalle fasi di creazione e test all'implementazione negli ambienti di produzione.

Anche se la prassi comune suggerisce di includere le persone nelle procedure operative più difficili, suggeriamo di automatizzare le procedure più difficili proprio per questo motivo.

Passaggi dell'implementazione

Per automatizzare le implementazioni ed eliminare le operazioni manuali, segui questi passaggi:

- Configura un repository di codice per archiviare il codice in modo sicuro: usa AWS CodeCommit per creare un repository sicuro basato su Git.
- Configura un servizio di integrazione continua per compilare il codice sorgente, eseguire i test e creare gli artefatti di implementazione: per impostare un progetto di compilazione per questo scopo, consulta Getting started with AWS CodeBuild using the console.
- Configura un servizio di implementazione che automatizzi le implementazioni delle applicazioni e gestisca la complessità degli aggiornamenti delle applicazioni senza dipendere da implementazioni manuali soggette a errori: AWS CodeDeploy automatizza le implementazioni del software su una varietà di servizi di calcolo, come Amazon EC2, AWS Fargate, AWS Lambda e i server locali. Per configurare questi passaggi, consulta Getting started with CodeDeploy.
- Configura un servizio di distribuzione continua che automatizzi le pipeline di rilascio per eseguire aggiornamenti più rapidi e affidabili di applicazioni e infrastrutture: prendi in considerazione AWS CodePipeline per automatizzare le pipeline di rilascio. Per maggiori dettagli, consulta CodePipeline tutorials.

Risorse

Best practice correlate:

- OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione
- OPS05-BP10 Automazione completa dell'integrazione e dell'implementazione
- OPS06-BP02 Implementazioni dei test
- OPS06-BP04 Automazione dei test e del rollback

Documenti correlati:

- Continuous Delivery of Nested AWS CloudFormation Stacks Using AWS CodePipeline
- Complete CI/CD with AWS CodeCommit, AWS CodeBuild, AWS CodeDeploy, and AWS CodePipeline
- APN Partner: partners that can help you create automated deployment solutions
- Marketplace AWS: products that can be used to automate your deployments

- Automate chat messages with webhooks.
- Amazon Builders' Library: Garantire la sicurezza del rollback durante le distribuzioni
- Amazon Builders' Library: Più velocità con una consegna continua
- What is AWS CodePipeline?
- What is CodeDeploy?
- AWS Systems Manager Patch Manager
- What is Amazon SES?
- What is Amazon Simple Notification Service?

Video correlati:

- AWS Summit 2019: CI/CD on AWS

Gestione degli errori

Domande

- REL 9. In che modo eseguire il backup dei dati?
- REL 10. Come si utilizza l'isolamento dei guasti per proteggere il carico di lavoro?
- REL 11. Come si progetta il carico di lavoro affinché resista ai guasti dei componenti?
- REL 12. Come si testa l'affidabilità?
- REL 13. Come si pianifica il disaster recovery o ripristino di emergenza?

REL 9. In che modo eseguire il backup dei dati?

Esegui il backup dei dati, delle applicazioni e della configurazione per soddisfare i tuoi requisiti relativi agli obiettivi di tempo di ripristino (recovery time objective, RTO) e agli obiettivi di punto di ripristino (recovery point objective, RPO).

Best practice

- REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o riproduzione dei dati dalle origini
- REL09-BP02 Protezione e crittografia dei backup

- REL09-BP03 Esecuzione del backup dei dati in automatico
- REL09-BP04 Ripristino periodico dei dati per verificare l'integrità e i processi di backup:

REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o riproduzione dei dati dalle origini

Scopri e usa le funzionalità di backup dei servizi per i dati e delle risorse utilizzati dal carico di lavoro. La maggior parte dei servizi offre funzionalità per eseguire il backup dei dati del carico di lavoro.

Risultato desiderato: capacità di identificare e classificare le origini dati in base alla criticità. Quindi, stabilisci una strategia per il recupero dei dati in base all'RPO. Questa strategia prevede il backup di queste origini dei dati o la possibilità di riprodurre i dati da altre origini. In caso di perdita di dati, la strategia implementata consente il recupero o la riproduzione dei dati entro i termini RPO e RTO definiti.

Fase di maturità del cloud: di base

Anti-pattern comuni:

- Mancata conoscenza di tutte le origini dei dati per il carico di lavoro e della loro criticità.
- Non si eseguono backup delle origini dei dati critiche.
- Esecuzione di backup solo di alcune origini dei dati senza utilizzare la criticità come criterio.
- Non esiste un RPO definito o la frequenza di backup non può soddisfare l'RPO.
- Nessuna valutazione della necessità di un backup o della possibilità di riprodurre i dati da altre origini.

Vantaggi dell'adozione di questa best practice: l'identificazione dei punti in cui sono necessari backup e l'implementazione di un meccanismo per la creazione di backup, o la riproduzione dei dati da un'origine esterna, migliorano la capacità di ripristinare e recuperare dati durante un'interruzione.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

Tutti i data store AWS offrono funzionalità di backup. Servizi come Amazon RDS e Amazon DynamoDB supportano inoltre il backup automatico che consente il ripristino point-in-time (PITR), grazie al quale è possibile ripristinare un backup in qualsiasi momento fino a cinque minuti o meno

rispetto all'ora corrente. Molti servizi AWS permettono di copiare backup in un'altra Regione AWS. AWS Backup è uno strumento che permette di centralizzare e automatizzare la protezione dei dati tra vari servizi AWS. AWS Elastic Disaster Recovery permette di copiare carichi di lavoro server completi e mantenere una protezione continua dei dati on-premise, tra zone di disponibilità o tra regioni con un obiettivo del punto di ripristino (RPO) misurato in secondi.

Amazon S3 può essere utilizzato come destinazione di backup per le origini dei dati gestite dal cliente e gestite da AWS. I servizi AWS come Amazon EBS, Amazon RDS e Amazon DynamoDB hanno funzionalità incorporate per creare i backup. È anche possibile utilizzare software di backup di terze parti.

È possibile eseguire il backup di dati on-premise nel Cloud AWS usando AWS Storage Gateway o AWS DataSync. È possibile usare bucket Amazon S3 per archiviare questi dati in AWS. Amazon S3 offre più livelli di archiviazione, come Amazon S3 Glacier o Deep Archive S3 Glacier, per ridurre i costi di archiviazione dei dati.

Potresti essere in grado di soddisfare le esigenze di recupero dei dati riproducendo i dati da altre origini. Ad esempio, potresti usare nodi di replica Amazon ElastiCache o repliche di lettura Amazon RDS per riprodurre i dati in caso di perdita del nodo primario. Nei casi in cui origini come questa possono essere usate per soddisfare l'obiettivo del punto di ripristino (RPO) e l'obiettivo del tempo di ripristino (RTO), un backup può non essere necessario. Come esempio aggiuntivo, se usi Amazon EMR, il backup del datastore HDFS può non essere necessario, purché sia possibile riprodurre i dati in Amazon EMR da Amazon S3.

Quando scegli una strategia di backup, devi considerare il tempo necessario per il ripristino dei dati. Il tempo necessario per il ripristino dei dati dipende dal tipo di backup (nel caso di una strategia di backup) o dalla complessità del meccanismo di riproduzione dei dati. Questo tempo deve rientrare nell'RTO per il carico di lavoro.

Passaggi dell'implementazione

1. Identifica tutte le origini dati per il carico di lavoro. I dati possono essere archiviati in diverse risorse, come database, volumi, file system, sistemi di registrazione e risorse di archiviazione di oggetti. Consulta la sezione Risorse per trovare i documenti correlati su diversi servizi AWS che offrono l'archiviazione di dati e sulle funzionalità offerte da questi servizi.
2. Classifica le origini dati in base alla criticità. I diversi set di dati avranno diversi livelli di criticità per un carico di lavoro e quindi diversi requisiti di resilienza. Ad esempio, alcuni dati possono essere critici e richiedere un RPO prossimo allo zero, mentre altri dati possono essere meno critici e

tollerare un RPO più elevato e una certa perdita di dati. Allo stesso modo, anche i diversi set di dati possono avere requisiti RTO diversi.

3. Usa AWS o servizi di terze parti per creare backup dei dati. AWS Backup è un servizio gestito che permette la creazione di backup di origini dati diverse in AWS. AWS Elastic Disaster Recovery gestisce la replica automatica dei dati in una Regione AWS con tempi inferiori al secondo. La maggior parte dei servizi AWS include anche funzionalità native per la creazione di backup. Marketplace AWS ha molte soluzioni che offrono anche queste funzionalità. Consulta la sezione Risorse di seguito per informazioni su come creare backup di dati da diversi servizi AWS.
4. Per i dati non sottoposti a backup, definisci un meccanismo di riproduzione dei dati. Puoi decidere di non eseguire il backup di dati riproducibili da altre origini per vari motivi. Potrebbe essere più conveniente riprodurre i dati dalle origini, quando necessario, piuttosto che creare un backup, dato che l'archiviazione dei backup può comportare dei costi. Un altro esempio è quello in cui il ripristino da un backup richiede più tempo rispetto alla riproduzione dei dati dalle origini, con conseguente violazione dell'RTO. In queste situazioni, è necessario considerare i compromessi e stabilire un processo ben definito per la riproduzione dei dati da queste origini quando è necessario il ripristino dei dati. Ad esempio, se hai caricato dati da Amazon S3 su un data warehouse (come Amazon Redshift) o su un cluster MapReduce (come Amazon EMR) per compiere analisi, ottieni un esempio pratico di riproduzione dati da oltre origini. Finché i risultati di queste analisi vengono archiviati o sono riproducibili, non subirai una perdita di dati a causa di un guasto nel data warehouse o nel cluster MapReduce. Altri esempi che possono essere riprodotti dalle origini includono le cache (ad esempio Amazon ElastiCache) o le repliche di lettura RDS.
5. Definisci una cadenza per il backup dei dati. La creazione di backup delle origini dei dati è un processo periodico e la frequenza deve dipendere dall'RPO.

Livello di impegno per il piano di implementazione: moderato.

Risorse

Best practice correlate:

REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati

REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino

Documenti correlati:

- Che cos'è AWS Backup?

- Che cos'è AWS DataSync?
- What is Volume Gateway? (Che cos'è il Gateway di volumi?)
- Partner APN: partner per il backup
- Marketplace AWS: prodotti che possono essere usati per il backup
- Snapshot Amazon EBS
- Backup in Amazon EFS
- Backup in Amazon FSx per Windows File Server
- Backup e ripristino di ElastiCache for Redis
- Creazione di shapshot di cluster di database in Neptune
- Creazione di uno snapshot DB
- Creazione di una regola EventBridge attivata in base a una pianificazione
- Replica tra regioni con Amazon S3
- AWS Backup da EFS a EFS
- Esportazione di dati di log in Amazon S3
- Gestione del ciclo di vita dell'applicazione
- Backup e ripristino on demand per DynamoDB
- Ripristino point-in-time (PITR) per DynamoDB
- Uso di snapshot di indici Amazon OpenSearch Service
- Che cos'è AWS Elastic Disaster Recovery?

Video correlati:

- AWS re:Invent 2021: Backup, ripristino di emergenza e protezione da ransomware con AWS
- Demo su AWS Backup: Backup tra account e tra regioni
- AWS re:Invent 2019: Approfondimento su AWS Backup, con Rackspace (STG341)

Esempi correlati:

- Well-Architected Lab: Implementazione della replica bidirezionale tra regioni per Amazon S3
- Well-Architected Lab: Esecuzione di test del backup e del ripristino di dati
- Well-Architected Lab: Backup e ripristino con failback per un carico di lavoro di analisi

- Well-Architected Lab: Ripristino di emergenza – Backup e ripristino

REL09-BP02 Protezione e crittografia dei backup

Controlla e rileva l'accesso ai backup tramite l'autenticazione e l'autorizzazione. Previeni e rileva se l'integrità dei dati dei backup è compromessa utilizzando la crittografia.

Anti-pattern comuni:

- Disporre di un accesso identico sia per i backup e l'automazione del ripristino sia per i dati.
- Non codificare i backup.

Vantaggi dell'adozione di questa best practice: la protezione dei backup previene la manomissione dei dati, mentre la crittografia dei dati impedisce l'accesso ai dati se questi vengono accidentalmente esposti.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

Controlla e rileva l'accesso ai backup tramite l'autenticazione e l'autorizzazione, ad esempio con AWS Identity and Access Management (IAM). Previeni e rileva se l'integrità dei dati dei backup è compromessa utilizzando la crittografia.

Amazon S3 supporta diversi metodi di crittografia dei dati archiviati. Utilizzando la crittografia lato server, Amazon S3 accetta anche dati non crittografati e li crittografa man mano che vengono memorizzati. Utilizzando la crittografia lato client, l'applicazione del carico di lavoro è responsabile della crittografia dei dati prima che vengano inviati a Amazon S3. Entrambi i metodi ti consentono di utilizzare AWS Key Management Service (AWS KMS) per creare ed archiviare la chiave di crittografia dei dati, oppure di utilizzarne una personalizzata (della quale sarai responsabile). Tramite AWS KMS puoi impostare delle policy utilizzando IAM per regolare l'accesso alle chiavi dei dati, oltre che ai dati privi di crittografia.

Per Amazon RDS, se hai scelto di crittografare i database, anche i backup verranno crittografati. I backup di DynamoDB sono sempre crittografati. Quando usi AWS Elastic Disaster Recovery, vengono crittografati tutti i dati in transito e a riposo. Con Elastic Disaster Recovery i dati a riposo possono essere crittografati tramite la chiave di crittografia dei volumi della crittografia predefinita in Amazon EBS o una chiave gestita dal cliente personalizzata.

Passaggi dell'implementazione

1. Utilizzo della crittografia su ciascuno dei datastore. Se i dati di origine sono crittografati, lo sarà anche il backup.
   - Usa la crittografia in Amazon RDS.. Puoi configurare la crittografia dei dati a riposo utilizzando AWS Key Management Service al momento della creazione di un'istanza RDS.
   - Usa la crittografia su volumi Amazon EBS.. Puoi configurare la crittografia predefinita o specificare una chiave univoca al momento della creazione del volume.
   - Usa la crittografia Amazon DynamoDB necessaria. DynamoDB esegue la crittografia di tutti i dati a riposo. Puoi utilizzare una chiave AWS KMS di proprietà di AWS o una chiave KMS gestita da AWS specificando una chiave archiviata nel tuo account.
   - Esegui la crittografia dei dati archiviati in Amazon EFS. Configura la crittografia al momento della creazione del file system.
   - Configura la crittografia nelle regioni di origine e di destinazione. Puoi configurare la crittografia dei dati a riposo in Amazon S3 utilizzando le chiavi archiviate in KMS, ma le chiavi sono specifiche per regione. Puoi specificare le chiavi di destinazione quando configuri la replica.
   - Scegli se usare la crittografia Amazon EBS predefinita o personalizzata per Elastic Disaster Recovery. Questa opzione esegue la crittografia dei dati a riposo replicati nei dischi della sottorete dell'area di staging e nei dischi replicati.
2. Implementazione delle autorizzazioni con privilegi minimi per accedere ai backup. Segui le best practice per limitare l'accesso a backup, snapshot e repliche in linea con le best practice per la sicurezza.

Risorse

Documenti correlati:

- Marketplace AWS: prodotti che possono essere usati per il backup
- Crittografia in Amazon EBS
- Amazon S3: protezione dei dati tramite crittografia
- Configurazione aggiuntiva della replica tra regioni: replica di oggetti creati con la crittografia lato server (SSE) usando chiavi di crittografia archiviate in AWS KMS
- Crittografia dei dati a riposo in DynamoDB
- Crittografia di risorse Amazon RDS
- Crittografia di dati e metadati in Amazon EFS
- Crittografia per i backup in AWS

- Gestione di tabelle crittografate
- Principio della sicurezza: Framework AWS Well-Architected
- Che cos'è AWS Elastic Disaster Recovery?

Esempi correlati:

- Well-Architected Lab: Implementazione della replica bidirezionale tra regioni per Amazon S3

REL09-BP03 Esecuzione del backup dei dati in automatico

Configura i backup in modo che vengano eseguiti automaticamente in base a una pianificazione periodica informata dall'Obiettivo del punto di ripristino (RPO) o dalle modifiche apportate al set di dati. I set di dati critici con bassi requisiti di perdita di dati devono essere sottoposti a backup automatico su base frequente, mentre i dati meno critici, per i quali è accettabile una certa perdita, possono essere sottoposti a backup meno frequenti.

Risultato desiderato: un processo di backup automatico che crea backup delle origini dati a una cadenza prestabilita.

Anti-pattern comuni:

- Eseguire i backup manualmente.
- Utilizzare risorse che dispongono di funzionalità di backup, ma non includere il backup nell'automazione.

Vantaggi dell'adozione di questa best practice: l'automazione dei backup verifica che i backup vengano eseguiti regolarmente in base all'RPO e invia avvisi in caso contrario.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

AWS Backup può essere utilizzato per creare backup automatici di varie origini dei dati AWS. Il backup delle istanze Amazon RDS può essere eseguito quasi ininterrottamente ogni cinque minuti e quello degli oggetti Amazon S3 quasi ininterrottamente ogni quindici minuti, consentendo il ripristino point-in-time (PITR) a un punto specifico della cronologia di backup. Per altre origini dei dati AWS, come volumi Amazon EBS, tabelle Amazon DynamoDB o file system Amazon FSx, AWS Backup può eseguire il backup automatico con una frequenza di un'ora. Questi servizi offrono anche funzionalità

di backup native. I servizi AWS che offrono il backup automatico con ripristino point-in-time (PITR) includono Amazon DynamoDB, Amazon RDS e Amazon Keyspaces (per Apache Cassandra). Questi servizi permettono il ripristino temporizzato in base a un momento specifico all'interno della cronologia dei backup. La maggior parte degli altri servizi di archiviazione di dati AWS offre la possibilità di programmare backup periodici, anche ogni ora.

Amazon RDS e Amazon DynamoDB offrono il backup continuo con ripristino point-in-time (PITR). Una volta abilitato, il controllo delle versioni in Amazon S3 è automatico. Puoi usare Amazon Data Lifecycle Manager per automatizzare la creazione, la copia e l'eliminazione di shapshot Amazon EBS. Può anche automatizzare la creazione, la copia, la rimozione e la cancellazione di Amazon Machine Images (AMI) con backup Amazon EBS e dei relativi snapshot Amazon EBS sottostanti.

AWS Elastic Disaster Recovery offre la replica a livello di blocco continua dall'ambiente di origine (on-premise o AWS) alla regione di ripristino di destinazione. Gli snapshot Amazon EBS point-in-time vengono creati e gestiti automaticamente dal servizio.

Per una visualizzazione centralizzata dell'automazione e della cronologia dei backup, AWS Backup fornisce una soluzione di backup completamente gestita basata su policy. Centralizza e automatizza il backup dei dati su più servizi AWS nel cloud e on-premise utilizzando AWS Storage Gateway.

Oltre a quella di controllo delle versioni, Amazon S3 offre tutte le funzioni di replica. L'intero bucket S3 può essere replicato automaticamente in un altro bucket in una Regione AWS diversa.

Passaggi dell'implementazione

1. Identifica le origini dati di cui al momento viene eseguito manualmente il backup. Per ulteriori informazioni, consulta REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o riproduzione dei dati dalle origini.
2. Determina l'RPO per il carico di lavoro. Per ulteriori informazioni, consulta REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati.
3. Usa una soluzione di backup automatica o un servizio gestito. AWS Backup è un servizio completamente gestito che semplifica la centralizzazione e l'automazione della protezione dei dati tra diversi servizi AWS, nel cloud e on-premise. Usando piani di backup in AWS Backup, crea regole che definiscano le risorse di cui eseguire il backup e la frequenza di creazione dei backup. Questa frequenza deve essere informata dall'RPO stabilito al punto 2. Per linee guida pratiche su come creare backup automatici con AWS Backup, consulta Well-Architected Lab: Test del backup e del ripristino di dati. La maggior parte dei servizi AWS di archiviazione dei dati offre funzionalità di backup native. Ad esempio, RDS può essere sfruttato per backup automatici con ripristino point-in-time (PITR).

4. Per le origini dati non supportate da una soluzione di backup automatica o da un servizio gestito, come le code di messaggi o le origini dati on-premise, valuta se usare una soluzione di terze parti affidabile per creare backup automatici. In alternativa, puoi creare un'automazione utilizzando la AWS CLI o gli SDK. Puoi usare funzioni AWS Lambda o AWS Step Functions per definire la logica necessaria per la creazione di un backup di dati e Amazon EventBridge per eseguirla in base a una frequenza determinata dall'RPO.

Livello di impegno per il piano di implementazione: basso.

Risorse

Documenti correlati:

- Partner APN: partner per il backup
- Marketplace AWS: prodotti che possono essere usati per il backup
- Creazione di una regola EventBridge attivata in base a una pianificazione
- Che cos'è AWS Backup?
- Che cos'è AWS Step Functions?
- Che cos'è AWS Elastic Disaster Recovery?

Video correlati:

- AWS re:Invent 2019: Approfondimento su AWS Backup, con Rackspace (STG341)

Esempi correlati:

- Well-Architected Lab: Esecuzione di test del backup e del ripristino di dati

REL09-BP04 Ripristino periodico dei dati per verificare l'integrità e i processi di backup:

Verifica che l'implementazione del processo di backup soddisfi gli obiettivi del tempo di ripristino (RTO) e gli obiettivi del punto di ripristino (RPO) eseguendo un test del ripristino.

Risultato desiderato: ripristino periodico dei dati dai backup tramite meccanismi ben definiti per garantire che il ripristino sia possibile entro l'obiettivo del tempo di ripristino (RTO) stabilito per il carico di lavoro. Verifica che il ripristino da un backup porti a una risorsa che contiene i dati originali

senza che questi siano danneggiati o inaccessibili e con una perdita di dati entro l'Obiettivo del punto di ripristino (RPO).

Anti-pattern comuni:

- Ripristino di un backup, ma senza eseguire query sui dati o recuperarli per verificare di poter usare il ripristino.
- Presupporre l'esistenza di un backup.
- Presupporre che il backup di un sistema sia pienamente operativo e che i dati possano essere recuperati da esso.
- Presupporre che il tempo di ripristino o di recupero dei dati da un backup rientri nell'RTO del carico di lavoro.
- Presupporre che i dati contenuti nel backup rientrino nell'RPO del carico di lavoro.
- Ripristino in base alle esigenze, senza usare un runbook o seguire una procedura automatica prestabilita.

Vantaggi dell'adozione di questa best practice: i test del ripristino dei backup permettono di verificare che i dati possano essere ripristinati senza timore che siano mancanti o danneggiati, che il ripristino e il recupero siano possibili in base all'RTO per il carico di lavoro e che un'eventuale perdita di dati rientri nell'RPO per il carico di lavoro.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

La verifica delle capacità di backup e ripristino aumenta la fiducia nella capacità di eseguire queste azioni durante un'interruzione. Ripristina periodicamente i backup in una nuova posizione ed esegui test per verificare l'integrità dei dati. Alcuni test comuni che devono essere eseguiti sono la verifica che tutti i dati siano disponibili, non siano danneggiati e siano accessibili e che un'eventuale perdita di dati rientri nell'RPO per il carico di lavoro. Questi test possono anche aiutare a verificare se i meccanismi di ripristino sono sufficientemente veloci per soddisfare l'RTO del carico di lavoro.

Con AWS, puoi creare un ambiente di test e ripristinare i backup per valutare le funzionalità RTO e RPO ed eseguire test sul contenuto e l'integrità dei dati.

Inoltre, Amazon RDS e Amazon DynamoDB consentono il ripristino point-in-time (PITR). Utilizzando il backup continuo, puoi ripristinare il set di dati allo stato in cui si trovava in una data e un'ora specificate.

tutti i dati siano disponibili, non siano danneggiati, siano accessibili e che qualsiasi perdita di dati rientri nell'RPO del carico di lavoro. Questi test possono anche aiutare a verificare se i meccanismi di ripristino sono sufficientemente veloci per soddisfare l'RTO del carico di lavoro.

AWS Elastic Disaster Recovery offre snapshot di ripristino point-in-time (RPIT) continui di volumi Amazon EBS. Durante la replica dei server di origine, vengono registrati gli stati point-in-time nel corso del tempo in base alla policy configurata. Elastic Disaster Recovery permette di verificare l'integrità di questi snapshot avviando istanze per scopi di test ed esercitazione senza reindirizzare il traffico.

Passaggi dell'implementazione

1. Identifica le origini dati di cui viene attualmente eseguito il backup e le posizioni in cui vengono archiviati i backup. Per le linee guida di implementazione, consulta REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o riproduzione dei dati dalle origini.
2. Definisci i criteri per la convalida dei dati per ogni origine dati. Tipi di dati differenti avranno proprietà diverse che potrebbero richiedere meccanismi di convalida diversi. Considera il modo in cui potrebbero essere convalidati questi dati prima di poterli utilizzare in produzione. Alcuni modi comuni per convalidare i dati sono l'uso delle loro proprietà dei dati e del backup, come il tipo di dati, il formato, la somma di controllo, la dimensione o la combinazione di questi elementi con una logica di convalida personalizzata. Ad esempio, può trattarsi di un confronto dei valori di checksum tra la risorsa ripristinata e l'origine dei dati al momento della creazione del backup.
3. Definisci l'RTO e l'RPO per il ripristino dei dati in base alle criticità dei dati. Per le linee guida di implementazione, consulta REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati.
4. Valuta la capacità di ripristino. Rivedi la strategia di backup e ripristino per capire se è in grado di soddisfare RTO e RPO e modifica la strategia se necessario. Usando la Centrale di resilienza AWS, puoi eseguire una valutazione del carico di lavoro. La valutazione esamina la configurazione dell'applicazione rispetto alle policy sulla resilienza e indica se gli obiettivi RTO e RPO possono essere raggiunti.
5. Esegui un ripristino di test usando i processi attualmente definiti nell'ambiente di produzione per il ripristino dei dati. Questi processi dipendono dal modo in cui è stato eseguito il backup dell'origine dei dati iniziale, dal formato e dalla posizione di archiviazione del backup stesso o dalla riproduzione dei dati da altre fonti. Ad esempio, se usi un servizio gestito come AWS Backup, può trattarsi di un semplice ripristino del backup in una nuova risorsa. Se hai usato AWS Elastic Disaster Recovery, puoi avviare un'esercitazione di ripristino.

6. Convalida il ripristino dei dati dalla risorsa ripristinata in base ai criteri definiti in precedenza per la convalida dei dati. I dati ripristinati e recuperati contengono il record o la voce più recente al momento del backup? Questi dati rientrano nell'RPO per il carico di lavoro?
7. Misura il tempo necessario per il ripristino e il recupero e confrontalo con l'RTO definito. Questo tempo deve rientrare nell'RTO per il carico di lavoro? Ad esempio, confronta i timestamp dell'inizio del processo di ripristino e del completamento della convalida del ripristino per calcolare la durata del processo. Tutte le chiamate API AWS includono un timestamp e queste informazioni sono disponibili in AWS CloudTrail. Sebbene queste informazioni possano fornire dettagli sull'inizio del processo di ripristino, la logica di convalida dovrebbe registrare il timestamp finale del completamento della convalida. Se usi un processo automatico, servizi come Amazon DynamoDB possono essere utili per archiviare queste informazioni. Inoltre, molti servizi AWS offrono una cronologia degli eventi che fornisce informazioni con data e ora in cui si sono verificate determinate azioni. In AWS Backup le attività di backup e ripristino sono note come processi e tali processi possono contenere informazioni sul timestamp come parte dei metadati, che possono essere usate per misurare il tempo necessario per il ripristino e il recupero.
8. Comunica agli stakeholder se la convalida dei dati non riesce o se il tempo necessario per il ripristino e il recupero supera l'RTO definito per il carico di lavoro. Nell'implementare l'automazione a questo scopo, come in questo lab, puoi usare servizi come Amazon Simple Notification Service (Amazon SNS) per inviare notifiche push come e-mail o SMS agli stakeholder. Questi messaggi possono anche essere pubblicati in applicazioni di messaggistica come Amazon Chime, Slack o Microsoft Teams o usati per creare attività come OpsItem usando OpsCenter di AWS Systems Manager.
9. Automatizza questo processo in modo che venga eseguito periodicamente Ad esempio, per automatizzare i processi di ripristino e recupero si possono utilizzare servizi come AWS Lambda o una State Machine in AWS Step Functions, mentre Amazon EventBridge può essere utilizzato per attivare periodicamente questo flusso di lavoro di automazione, come mostrato nel diagramma di architettura sottostante. Per altre informazioni, consulta Automazione della convalida del ripristino di dati con AWS Backup. Inoltre, questo Well-Architected Lab permette di acquisire esperienza pratica su un modo per implementare l'automazione per diverse fasi presentate qui.

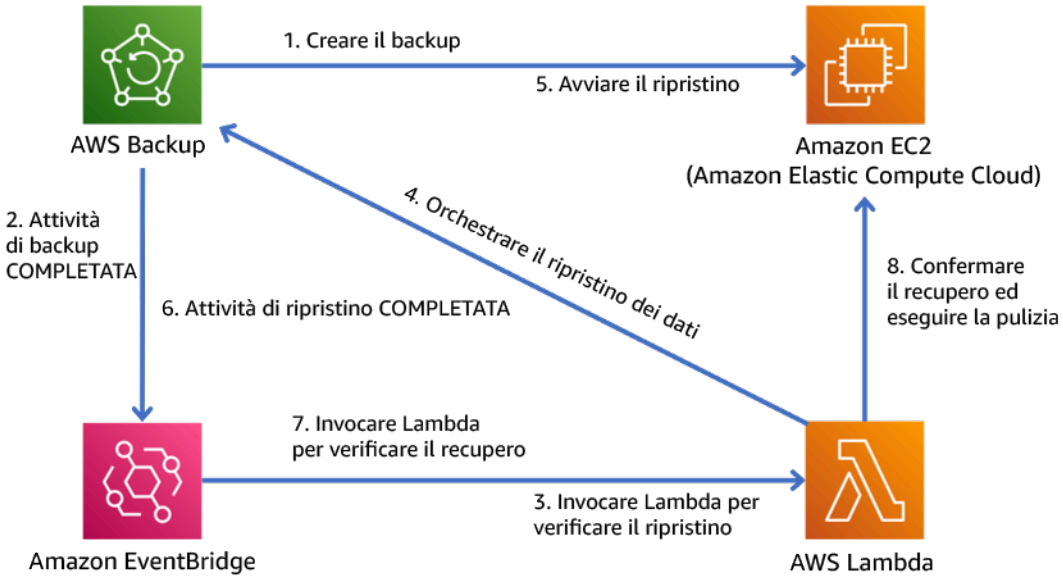


Figura 9. Processo di backup e ripristino automatico

Livello di impegno per il piano di implementazione: da moderato a elevato, a seconda della complessità dei criteri di convalida.

Risorse

Documenti correlati:

- Automazione della convalida del ripristino di dati con AWS Backup
- Partner APN: partner per il backup
- Marketplace AWS: prodotti che possono essere usati per il backup
- Creazione di una regola EventBridge attivata in base a una pianificazione
- Backup e ripristino on demand per DynamoDB
- Che cos'è AWS Backup?
- Che cos'è AWS Step Functions?
- Che cos'è AWS Elastic Disaster Recovery?
- AWS Elastic Disaster Recovery

Esempi correlati:

- Well-Architected Lab: Esecuzione di test del backup e del ripristino di dati

REL 10. Come si utilizza l'isolamento dei guasti per proteggere il carico di lavoro?

Le barriere per l'isolamento dei guasti limitano l'effetto di un errore all'interno di un carico di lavoro a un numero limitato di componenti. I componenti al di fuori della barriera non subiscono gli effetti del guasto. Utilizzando più barriere per l'isolamento dei guasti, puoi limitare l'impatto sul carico di lavoro.

Best practice

- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL10-BP02 Selezione delle posizioni appropriate per la tua implementazione multiposizione
- REL10-BP03 Ripristino automatico dei componenti vincolati a una singola posizione
- REL10-BP04 Utilizzo di architetture a scomparti per limitare la portata dell'impatto

REL10-BP01 Implementazione del carico di lavoro in diversi luoghi

Distribuisci i dati e le risorse del carico di lavoro su più zone di disponibilità o, se necessario, su diverse Regioni AWS. Questi luoghi possono essere diversi a seconda delle necessità.

Uno dei principi fondamentali per la progettazione di servizi su AWS è l'eliminazione di singoli punti di errore nell'infrastruttura fisica sottostante. Questo ci spinge a creare software e sistemi che utilizzano più zone di disponibilità e sono resistenti al fallimento di una singola zona. Allo stesso modo, i sistemi sono costruiti per resistere ai guasti di un singolo nodo di calcolo, singolo volume di archiviazione o singola istanza di un database. Quando si costruisce un sistema che si basa su componenti ridondanti, è importante garantire che i componenti funzionino in modo indipendente e, nel caso delle Regioni AWS, in modo autonomo. I vantaggi ottenuti dai calcoli di disponibilità teorica con componenti ridondanti sono validi solo se questo continua a essere vero.

Zone di disponibilità (AZ)

Le Regioni AWS sono composte da almeno due zone di disponibilità progettate per essere indipendenti. Ogni zona di disponibilità è separata da una distanza fisica significativa da altre zone per evitare scenari di guasto correlati, dovuti a rischi ambientali come incendi, inondazioni e tornado. Ogni zona di disponibilità ha anche un'infrastruttura fisica indipendente: connessioni dedicate di alimentazione di rete, fonti di alimentazione di backup autonome, servizi meccanici indipendenti e connettività di rete indipendente all'interno e all'esterno della zona di disponibilità. Questa struttura limita gli errori di uno qualsiasi di questi sistemi alla sola AZ interessata. Nonostante siano geograficamente separate, le zone di disponibilità sono situate nella stessa area regionale, il che consente una rete a velocità di trasmissione effettiva elevata e bassa latenza. L'intera Regione AWS (in tutte le zone di disponibilità, costituite da più data center fisicamente indipendenti) può

essere trattata come un unico obiettivo logico di implementazione per il carico di lavoro, compresa la possibilità di replicare i dati in modo sincrono (ad esempio, tra i database). Ciò ti consente di utilizzare le zone di disponibilità in una configurazione attiva/attiva o attiva/standby.

Le zone di disponibilità sono indipendenti e pertanto la disponibilità del carico di lavoro aumenta quando il carico di lavoro è progettato per utilizzare più zone di disponibilità. Alcuni servizi AWS (tra cui il piano dati dell'istanza Amazon EC2) sono implementati come servizi strettamente zonali nei quali hanno un destino condiviso con la zona di disponibilità in cui si trovano. Le istanze Amazon EC2 nelle altre AZ non saranno, tuttavia, interessate e continueranno a funzionare. Allo stesso modo, se un errore in una zona di disponibilità causa l'errore di un database Amazon Aurora, un'istanza Aurora di lettura-replica in una AZ non interessata può essere automaticamente promossa a primaria. I servizi regionali AWS, ad esempio Amazon DynamoDB, utilizzano internamente più zone di disponibilità in una configurazione attiva/attiva per raggiungere gli obiettivi di progettazione della disponibilità per quel servizio, senza che sia necessario configurare il posizionamento delle AZ.

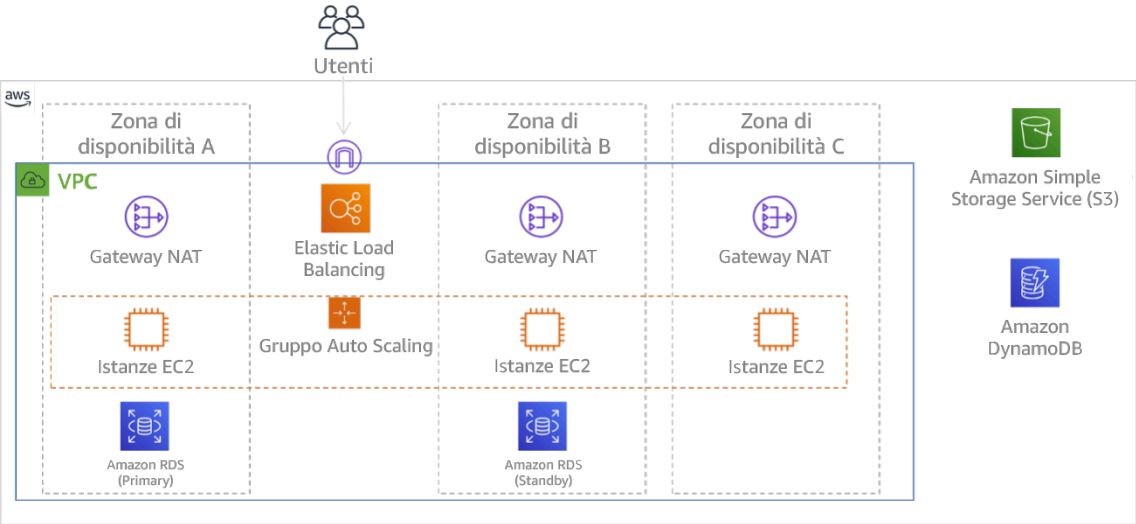


Figura 9. Architettura multi-livello distribuita su tre zone di disponibilità. Tieni presente che Amazon S3 e Amazon DynamoDB sono sempre Multi-AZ automaticamente. L'ELB viene inoltre distribuito in tutte e tre le zone.

Mentre i piani di controllo AWS in genere offrono la possibilità di gestire le risorse all'interno dell'intera Regione (più zone di disponibilità), alcuni piani di controllo (inclusi Amazon EC2 ed Amazon EBS) hanno la capacità di filtrare i risultati per una singola zona di disponibilità. Con questo approccio, la richiesta viene elaborata solo nella zona di disponibilità specificata, riducendo l'esposizione all'interruzione in altre zone di disponibilità. Questo esempio di AWS CLI illustra come ottenere informazioni su un'istanza Amazon EC2 dalla sola zona di disponibilità us-east-2c:

```
AWS ec2 describe-instances --filters Name=availability-zone,Values=us-east-2c
```

Zone locali AWS

Le Zone locali AWS agiscono in modo simile alle zone di disponibilità nella rispettiva Regione AWS, in quanto possono essere selezionate come ubicazione di posizionamento per le risorse AWS zonali come le sottoreti e le istanze EC2. Ciò che le rende speciali è che non si trovano nella Regione AWS associata, ma vicino a grandi popolazioni, settori e centri IT in cui al momento non esiste alcuna Regione AWS. Tuttavia, mantengono una connessione sicura e a larghezza di banda elevata tra i carichi di lavoro locali nella zona locale e quelli in esecuzione nella Regione AWS. È consigliabile utilizzare le Zone locali AWS per implementare i carichi di lavoro più vicini agli utenti per requisiti a bassa latenza.

Amazon Global Edge Network

Amazon Global Edge Network è costituito da posizioni edge in città di tutto il mondo. Amazon CloudFront utilizza questa rete per fornire contenuti agli utenti finali con una latenza inferiore. AWS Global Accelerator consente di creare gli endpoint del carico di lavoro in queste posizioni edge per fornire l'onboarding alla rete globale AWS vicino agli utenti. Amazon API Gateway permette agli endpoint API ottimizzati per l'edge che utilizzano una distribuzione CloudFront di facilitare l'accesso dei clienti attraverso la posizione edge più vicina.

Regioni AWS

Le Regioni AWS sono progettate per essere autonome; pertanto, per utilizzare un approccio multi-regione, puoi implementare copie dedicate dei servizi in ciascuna Regione.

Un approccio multi-regione è comune per le strategie di ripristino di emergenza per raggiungere gli obiettivi di ripristino quando si verificano eventi unici su larga scala. Consulta Pianificazione per il disaster recovery (DR) per ulteriori informazioni su queste strategie. Qui, tuttavia, si focalizza l'attenzione sulla disponibilità, che cerca di fornire un obiettivo medio di operatività nel tempo. Per gli obiettivi di alta disponibilità, un'architettura multi-regione sarà generalmente progettata per essere attiva/attiva, dove ogni copia del servizio (nelle rispettive Regioni) è attiva (serve le richieste).

> **Consiglio**
>
> Gli obiettivi di disponibilità per la maggior parte dei carichi di lavoro possono essere soddisfatti utilizzando una strategia multi-AZ all'interno di una singola Regione AWS.

Considera le architetture multi-regione solo quando i carichi di lavoro hanno requisiti di disponibilità estremi o altri obiettivi aziendali che richiedono un'architettura multi-regione.

AWS offre ai clienti la possibilità di gestire servizi in più Regioni. Ad esempio, AWS fornisce una replica continua e asincrona dei dati utilizzando la replica Amazon Simple Storage Service (Amazon S3), le repliche di lettura Amazon RDS (incluse le repliche di lettura Aurora) e le tabelle globali Amazon DynamoDB. Con la replica continua, le versioni dei dati sono disponibili per un uso quasi immediato in ogni Regione attiva.

Utilizzando AWS CloudFormation, puoi definire l'infrastruttura e implementarla in modo coerente sugli Account AWS e sulle Regioni AWS. Invece, AWS CloudFormation StackSets estende questa funzionalità consentendo di creare, aggiornare o eliminare stack AWS CloudFormation su più account e regioni con un'unica operazione. Per le implementazioni di istanza Amazon EC2, si utilizza un'immagine AMI (Amazon Machine Image) per fornire informazioni quali la configurazione hardware e il software installato. È possibile implementare una pipeline di Amazon EC2 Image Builder che crea le AMI necessarie e le copia nelle regioni attive. Ciò garantisce che le Golden AMI abbiano tutto ciò che serve per implementare e dimensionare il carico di lavoro in ogni nuova regione.

Per instradare il traffico, sia Amazon Route 53 sia AWS Global Accelerator abilitano la definizione di criteri che determinano quali utenti indirizzare a ogni endpoint regionale attivo. Con Global Accelerator imposti un valore di traffico per controllare la percentuale di traffico diretta a ciascun endpoint dell'applicazione. Route 53 supporta questo approccio percentuale e anche diverse altre policy disponibili, tra cui quelle basate sulla geoprossimità e sulla latenza. Global Accelerator sfrutta automaticamente la vasta rete di server edge AWS per convogliare il traffico verso la dorsale di rete AWS il prima possibile, con conseguente riduzione delle latenze delle richieste.

Tutte queste capacità operano in modo da preservare l'autonomia di ogni Regione. Ci sono pochissime eccezioni a questo approccio, inclusi i nostri servizi che forniscono distribuzione edge globale (ad esempio Amazon CloudFront e Amazon Route 53), insieme al piano di controllo per il servizio AWS Identity and Access Management (IAM). La maggior parte dei servizi opera interamente all'interno di una singola Regione.

Data center in locale

Per i carichi di lavoro eseguiti in un data center on-premise, puoi progettare un'esperienza ibrida quando possibile. AWS Direct Connect fornisce una connessione di rete dedicata dalla tua sede ad AWS che consente l'esecuzione in entrambi.

Un'altra opzione è quella di eseguire l'infrastruttura AWS e i servizi on-premise utilizzando AWS Outposts. AWS Outposts è un servizio completamente gestito che estende l'infrastruttura AWS, i servizi AWS, le API e gli strumenti al tuo data center. La stessa infrastruttura hardware utilizzata nel Cloud AWS viene installata nel data center. AWS Outposts è, quindi, connesso alla Regione AWS più vicina. Puoi quindi utilizzare AWS Outposts per supportare i carichi di lavoro che hanno requisiti di bassa latenza o di elaborazione dei dati locali.

Livello di rischio associato se questa best practice non fosse adottata: Alta

Guida all'implementazione

- Utilizza zone di disponibilità multiple e Regioni AWS. Distribuisci i dati e le risorse del carico di lavoro su più zone di disponibilità o, se necessario, su diverse Regioni AWS. Questi luoghi possono essere diversi a seconda delle necessità.
  - I servizi regionali sono distribuiti intrinsecamente in zone di disponibilità.
    - Sono inclusi Amazon S3, Amazon DynamoDB e AWS Lambda (se non collegati a un VPC)
  - Distribuisci il tuo container, istanza e carichi di lavoro basati su funzioni in più zone di disponibilità. Utilizza datastore multi-zona, inclusi sistemi di cache. Utilizza le funzionalità di dimensionamento automatico EC2, posizionamento di attività ECS, configurazione della funzione AWS Lambda in esecuzione nel tuo VPC e i cluster ElastiCache.
    - Utilizza sottoreti che sono in zone di disponibilità separate nella distribuzione di gruppi Auto Scaling.
      - Esempio: distribuzione di istanze in più zone di disponibilità
      - Strategie di posizionamento dei processi di Amazon ECS
      - Configurazione di una funzione AWS Lambda per accedere alle risorse in un Amazon VPC
      - Scelta di regioni e zone di disponibilità
    - Utilizza sottoreti in zone di disponibilità separate quando distribuisci gruppi Auto Scaling.
      - Esempio: distribuzione di istanze in più zone di disponibilità
    - Utilizza parametri di posizionamento attività ECS, specificando i gruppi di sottorete DB.
      - Strategie di posizionamento dei processi di Amazon ECS
    - Utilizza sottoreti in più zone di disponibilità quando configuri una funzione da eseguire nel tuo VPC.
      - Configurazione di una funzione AWS Lambda per accedere alle risorse in un Amazon VPC
    - Utilizza più zone di disponibilità con cluster ElastiCache.
      - Scelta di regioni e zone di disponibilità

- Se il carico di lavoro deve essere implementato in più Regioni, scegli una strategia multi-regione. La maggior parte delle esigenze di affidabilità può essere soddisfatta all'interno di una singola Regione AWS utilizzando una strategia a più zone di disponibilità. Quando necessario, utilizza una strategia multi-Regione per soddisfare le tue esigenze aziendali.
  - AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (Modelli architetturali per applicazioni attive-attive su più Regioni) (ARC209-R2)
    - Il backup in un'altra Regione AWS può garantire ulteriormente che i dati saranno disponibili quando necessario.
    - Alcuni carichi di lavoro hanno requisiti normativi che prevedono l'utilizzo di una strategia multi-regione.
- Valuta AWS Outposts per il tuo carico di lavoro. Se il carico di lavoro richiede bassa latenza nel data center locale o ha requisiti di elaborazione dei dati locali. In tal caso esegui l'infrastruttura e i servizi AWS on-premise utilizzando AWS Outposts.
  - Che cos'è AWS Outposts?
- Stabilisci se le Zone locali AWS ti aiutano a fornire il servizio ai tuoi utenti. Se hai requisiti di bassa latenza, verifica se le Zone locali AWS si trovano vicino ai tuoi utenti. Se sì, utilizzale per implementare carichi di lavoro più vicini a tali utenti.
  - Domande frequenti sulle Zone locali AWS

Risorse

Documenti correlati:

- Infrastruttura globale di AWS
- Domande frequenti sulle Zone locali AWS
- Strategie di posizionamento dei processi di Amazon ECS
- Scelta di regioni e zone di disponibilità
- Esempio: distribuzione di istanze in più zone di disponibilità
- Tabelle globali: replica multi-regione con DynamoDB
- Using Amazon Aurora global databases (Utilizzo di database Amazon Aurora globali)
- Creating a Multi-Region Application with AWS Services blog series (Creazione di un'applicazione multi-regione con la serie di blog sui servizi AWS)
- Che cos'è AWS Outposts?

Video correlati:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (Modelli architetturali per applicazioni attive-attive su più Regioni) (ARC209-R2)
- AWS re:Invent 2019: Innovation and operation of the AWS global network infrastructure (Innovazione e gestione dell'infrastruttura di rete globale AWS) (NET339)

REL10-BP02 Selezione delle posizioni appropriate per la tua implementazione multiposizione

Risultato desiderato

Per ottenere un'elevata disponibilità, distribuisci sempre (quando possibile) i componenti del carico di lavoro in più zone di disponibilità (AZ), come illustrato nella Figura 10. Per i carichi di lavoro con requisiti di resilienza estremi, valuta attentamente le opzioni per un'architettura multiregione.

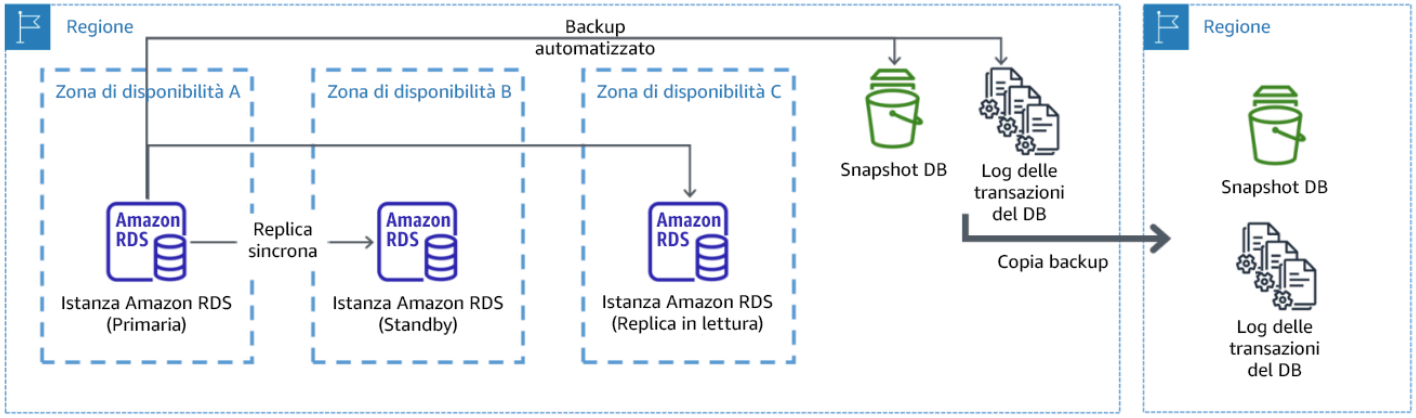


Figura 10: Distribuzione resiliente di un database multi-AZ con backup in un'altra regione AWS

Anti-pattern comuni

- Scelta di progettare un'architettura multi-regione quando un'architettura multi-AZ soddisferebbe i requisiti.
- Non si tiene conto delle dipendenze tra i componenti dell'applicazione se i requisiti di resilienza e multi-sede differiscono tra questi componenti.

Vantaggi dell'adozione di questa best practice

Per la resilienza, devi utilizzare un approccio che costruisca livelli di difesa. Un livello protegge dalle interruzioni più piccole e più comuni costruendo un'architettura ad alta disponibilità utilizzando più

AZ. Un altro livello di difesa è destinato a proteggere da eventi rari come disastri naturali diffusi e interruzioni a livello regionale. Questo secondo livello implica l'architettura dell'applicazione in modo che si estenda su più Regioni AWS.

- La differenza tra una disponibilità del 99,5% e una del 99,99% è di oltre 3,5 ore al mese. La disponibilità prevista di un carico di lavoro può raggiungere i "quattro nove" solo se si trova in più AZ.
- Eseguendo il carico di lavoro in più AZ, puoi isolare gli errori di alimentazione, raffreddamento e rete e la maggior parte dei disastri naturali come incendi e inondazioni.
- L'implementazione di una strategia multi-regione per il tuo carico di lavoro aiuta a proteggerlo da disastri naturali diffusi che colpiscono un'ampia regione geografica di un paese o da guasti tecnici di portata regionale. Tieni presente che l'implementazione di un'architettura multi-regione può essere molto complessa e di solito non è necessaria per la maggior parte dei carichi di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: Alta

Guida all'implementazione

Per un evento disastroso basato sull'interruzione o la perdita parziale di una zona di disponibilità, l'implementazione di un carico di lavoro a disponibilità elevata in più zone di disponibilità all'interno di una singola Regione AWS aiuta a mitigare i disastri naturali e tecnici. Ogni Regione AWS è composta da più zone di disponibilità, ciascuna isolata dagli errori nelle altre zone e separate da una distanza significativa. Tuttavia, per un evento di disastro che include il rischio di perdere più componenti della zona di disponibilità, che si trovano a una distanza significativa l'uno dall'altro, è necessario implementare opzioni di ripristino di emergenza per mitigare gli errori di portata regionale. Per i carichi di lavoro che richiedono un'estrema resilienza (infrastrutture critiche, applicazioni sanitarie, infrastrutture di sistemi finanziari e così via), può essere necessaria una strategia multi-regione.

Passaggi dell'implementazione

1. Valutare il carico di lavoro e determinare se le esigenze di resilienza possono essere soddisfatte da un approccio multi-AZ (Regione AWS singola) o se richiedono un approccio multi-regione. L'implementazione di un'architettura multi-regione per soddisfare questi requisiti introdurrà un'ulteriore complessità, quindi considera attentamente il tuo caso d'uso e i suoi requisiti. I requisiti di resilienza possono quasi sempre essere soddisfatti utilizzando un singolo Regione AWS. Per stabilire se è necessario utilizzare più Regioni, considera i seguenti possibili requisiti:
   a. Ripristino di emergenza: per un evento disastroso basato sull'interruzione o la perdita parziale di una zona di disponibilità, l'implementazione di un carico di lavoro a disponibilità elevata in più

zone di disponibilità all'interno di una singola Regione AWS aiuta a mitigare i disastri naturali e tecnici. In caso di eventi disastrosi che comportano il rischio di perdere più componenti della zone di disponibilità, che si trovano a una distanza significativa l'uno dall'altro, è necessario implementare il ripristino di emergenza in più regioni per mitigare i disastri naturali o gli errori tecnici di portata regionale.

b. Alta disponibilità: è possibile utilizzare un'architettura multi-regione (utilizzando più AZ in ogni regione) per ottenere una disponibilità superiore a quattro 9 (> 99,99%).

c. Localizzazione delle risorse: quando si distribuisce un carico di lavoro a un pubblico globale, è possibile distribuire stack localizzati in diverse Regioni AWS per servire il pubblico di quelle regioni. La localizzazione può includere la lingua, la valuta e i tipi di dati memorizzati.

d. Prossimità agli utenti: quando si distribuisce un carico di lavoro a un pubblico globale, è possibile ridurre la latenza distribuendo gli stack alle regioni Regioni AWS in prossimità degli utenti finali.

e. Posizione fisica dei dati: alcuni carichi di lavoro sono soggetti a requisiti di residenza dei dati, in base ai quali i dati di determinati utenti devono rimanere all'interno dei confini di un determinato Paese. In base alla normativa in questione, è possibile scegliere di distribuire un intero stack o solo i dati nella Regione AWS all'interno di tali confini.

2. Ecco alcuni esempi di funzionalità multi-AZ fornite dai servizi AWS:

a. Per proteggere i carichi di lavoro che utilizzano EC2 o ECS, è necessario distribuire un Elastic Load Balancer davanti alle risorse di calcolo. Elastic Load Balancing quindi fornisce la soluzione per rilevare le istanze nelle zone non integre e instradare il traffico verso quelle integre.

   i. Nozioni di base su Application Load Balancers

   ii. Nozioni di base su Network Load Balancer

b. Nel caso di istanze EC2 che eseguono software commerciale pronto all'uso e che non supportano il bilanciamento del carico, puoi ottenere una forma di tolleranza ai guasti implementando una metodologia di ripristino di emergenza multi-AZ.

   i. the section called “REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino”

c. Per le attività Amazon ECS, distribuire il servizio in modo uniforme su tre AZ per ottenere un equilibrio tra disponibilità e costi.

   i. Amazon ECS availability best practices | Containers (Best practice di disponibilità ECS | Container)

d. Per non Aurora Amazon RDS, puoi scegliere multi-AZ come opzione di configurazione. In caso di errore dell'istanza del database primario, Amazon RDS promuove automaticamente

un database standby per ricevere il traffico in un'altra zona di disponibilità. Puoi inoltre creare repliche di lettura multi-regione per migliorare la resilienza.

   i. Implementazioni Multi-AZ Amazon RDS

   ii. Creazione di una replica di lettura in un'altra Regione AWS

3. Ecco alcuni esempi di funzionalità multi-AZ fornite dai servizi AWS:

   a. Per i carichi di lavoro Amazon S3 in cui la disponibilità multi-AZ è fornita automaticamente dal servizio, considera i punti di accesso multi-regione se è necessaria un'implementazione multi-regione.

      i. Punti di accesso multi-regione in Amazon S3

   b. Per le tabelle DynamoDB in cui la disponibilità multi-AZ è fornita automaticamente dal servizio, è possibile convertire facilmente le tabelle esistenti in tabelle globali per sfruttare più regioni.

      i. Convert Your Single-Region Amazon DynamoDB Tables to Global Tables (Convertire le tabelle Amazon DynamoDB di una singola regione in tabelle globali)

   c. Se il carico di lavoro è gestito da Application Load Balancers o da Network Load Balancer, utilizza AWS Global Accelerator per migliorare la disponibilità dell'applicazione indirizzando il traffico verso più regioni che contengono endpoint integri.

      i. Endpoints for standard accelerators in AWS Global Accelerator - AWS Global Accelerator (Endpoint per acceleratori standard in AWS Global Accelerator) (amazon.com)

   d. Per le applicazioni che sfruttano AWS EventBridge, considera i bus tre regioni per inoltrare gli eventi ad altre regioni selezionate.

      i. Sending and receiving Amazon EventBridge events between Regioni AWS (Invio e ricezione di eventi Amazon EventBridge tra regioni AWS)

   e. Per i database Amazon Aurora, considera i database globali Aurora, che si estendono su più regioni AWS. I cluster esistenti possono essere modificati per aggiungere anche nuove Regioni.

      i. Nozioni di base sui database globali Amazon Aurora

   f. Se il carico di lavoro include chiavi di crittografia AWS Key Management Service (AWS KMS), valuta se le chiavi multi-regione sono adatte all'applicazione.

      i. Chiavi multi-regione in AWS KMS

   g. Per altre funzionalità del servizio AWS, vedi questa serie di blog su Creating a Multi-Region Application with AWS Services series (Creazione di un'applicazione multi-regione con la serie di servizi AWS)

Livello di impegno per il piano di implementazione: da moderato ad alto

Risorse

Documenti correlati:

- Creating a Multi-Region Application with AWS Services series (Creazione di un'applicazione multi-regione con la serie di servizi AWS)
- Disaster Recovery (DR) Architecture on AWS, Part IV: Multi-site Active/Active (Architettura di ripristino di emergenza su AWS, parte IV: attiva/attiva multi-sito)
- Infrastruttura globale di AWS
- Domande frequenti su AWS Local Zones
- Disaster Recovery (DR) Architecture on AWS, Part I: Strategies for Recovery in the Cloud (Architettura di ripristino di emergenza su AWS parte I: strategie per il ripristino nel cloud)
- Il ripristino di emergenza è differente nel cloud
- Tabelle globali: replica multi-regione con DynamoDB

Video correlati:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (Modelli di architettura per applicazioni attive-attive multiregione) (ARC209-R2)
- Auth0: architettura ad alta disponibilità multi-Regione che raggiunge più di 1,5 miliardi di accessi al mese con failover automatico

Esempi correlati:

- Disaster Recovery (DR) Architecture on AWS, Part I: Strategies for Recovery in the Cloud (Architettura di ripristino di emergenza su AWS parte I: strategie per il ripristino nel cloud)
- DTCC raggiunge livelli di resilienza superiori a quelli che raggiunge on-premise
- Expedia Group utilizza un'architettura multi-regione, a più zone di disponibilità con un servizio DNS proprietario per aggiungere resilienza alle applicazioni
- Uber: ripristino di emergenza per Kafka multi-Regione
- Netflix: attivo-attivo per la resilienza multi-regione
- Come costruiamo la posizione fisica dei dati per Atlassian Cloud
- Intuit TurboTax funziona in due regioni

REL10-BP03 Ripristino automatico dei componenti vincolati a una singola posizione

Se i componenti del carico di lavoro possono essere eseguiti in una sola zona di disponibilità o in un data center on-premise, devi rendere possibile la ricostruzione completa del carico di lavoro in base agli obiettivi di ripristino definiti.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Se, a causa di vincoli tecnologici, non è possibile seguire le linee guida per distribuire il carico di lavoro in più posizioni, è necessario implementare un percorso alternativo mirato alla resilienza. È necessario automatizzare la possibilità di ricreare l'infrastruttura necessaria, ridistribuire le applicazioni e ricreare i dati necessari per questi casi.

Ad esempio, Amazon EMR lancia tutti i nodi per un determinato cluster nella stessa zona di disponibilità: eseguire un cluster nella stessa zona migliora le prestazioni dei flussi di lavoro poiché fornisce una velocità di accesso ai dati più elevata. Se questo componente è necessario per la resilienza del carico di lavoro, è necessario disporre di un modo per implementare nuovamente il cluster e i relativi dati. Inoltre, per Amazon EMR è necessario effettuare il provisioning della ridondanza in modi diversi dall'utilizzo di Multi-AZ. Puoi effettuare il provisioning di più nodi. Usando il file system EMR (EMRFS), i dati in EMR possono essere ripristinati in Amazon S3, che a sua volta può essere replicato tra più zone di disponibilità o Regioni AWS.

Analogamente, Amazon Redshift per impostazione predefinita effettua il provisioning del cluster in una zona di disponibilità casuale all'interno della Regione AWS selezionata. Tutti i nodi del cluster vengono assegnati nella stessa zona.

Per carichi di lavoro basati su server stateful implementati in un data center on-premise, puoi usare AWS Elastic Disaster Recovery per proteggerli in AWS. Se il carico di lavoro è già ospitato in AWS, puoi usare Elastic Disaster Recovery per proteggerlo in una zona di disponibilità o regione alternativa. Elastic Disaster Recovery usa la replica a livello di blocco continua in un'area di staging leggera per fornire il ripristino rapido e affidabile di applicazioni on-premise e basate sul cloud.

Passaggi dell'implementazione

1. Implementa l'autoriparazione. Distribuisci le tue istanze o container utilizzando, quando possibile, il ridimensionamento automatico. Se non è possibile utilizzare il ridimensionamento automatico, utilizza il ripristino automatico per istanze EC2 o implementa l'automazione di autoriparazione in base agli eventi del ciclo di vita di container Amazon EC2 o ECS.

- Usa gruppi con Amazon EC2 Auto Scaling per carichi di lavoro in istanze e container che non abbiano requisiti di un singolo indirizzo IP, indirizzo IP privato o indirizzo IP elastico per le istanze e di metadati delle istanze.
  - È possibile usare i dati utente del modello di avvio per implementare l'automazione per la riparazione automatica della maggior parte dei carichi di lavoro.
- Usa il ripristino automatico di istanze Amazon EC2 per i carichi di lavoro che richiedono un singolo indirizzo IP, indirizzo IP privato o indirizzo IP elastico per le istanze e metadati delle istanze.
  - Il ripristino automatico invierà avvisi sullo stato del ripristino a un argomento SNS quando viene rilevato l'errore dell'istanza.
- Usa eventi del ciclo di vita delle istanze Amazon EC2 o eventi Amazon ECS per automatizzare la riparazione automatica quando il dimensionamento automatico o il recupero in EC2 non sono opzioni valide.
  - Utilizza gli eventi per invocare l'automazione che riparerà il tuo componente secondo la logica di processo richiesta.
- Proteggi i carichi di lavoro stateful limitati a una singola posizione usando AWS Elastic Disaster Recovery.

Risorse

Documenti correlati:

- Eventi Amazon ECS
- Hook del ciclo di vita Amazon EC2 Auto Scaling
- Recover your instance.
- Scalabilità automatica del servizio
- Che cos'è Amazon EC2 Auto Scaling?
- AWS Elastic Disaster Recovery

REL10-BP04 Utilizzo di architetture a scomparti per limitare la portata dell'impatto

Implementa architetture a scomparti (note anche come architetture basate su celle) per limitare l'effetto di un guasto all'interno di un carico di lavoro a un numero ridotto di componenti.

Risultato desiderato: un'architettura basata su celle usa più istanze isolate di un carico di lavoro, in cui ogni istanza è nota come cella. Ogni cella è indipendente, non condivide lo stato con altre celle e gestisce un sottoinsieme delle richieste complessive del carico di lavoro. Questo approccio riduce il possibile impatto di un errore, ad esempio un aggiornamento software non valido, a una singola cella e alle richieste elaborate. Se un carico di lavoro usa 10 celle per gestire 100 richieste e si verifica un errore, il 90% delle richieste complessive non sarà interessato dall'errore.

Anti-pattern comuni:

- Aumento illimitato delle celle.
- Applicazione di aggiornamenti o implementazioni del codice in tutte le celle contemporaneamente.
- Condivisione dello stato dei componenti tra celle (con l'eccezione del livello di instradamento).
- Aggiunta di logica di business o instradamento complessa al livello di instradamento.
- Le interazioni tra celle non sono ridotte al minimo.

Vantaggi dell'adozione di questa best practice: con un'architettura basata su celle, molti tipi comuni di errore sono limitati alla cella stessa, per un ulteriore isolamento degli errori. Queste limitazioni possono fornire resilienza a determinati tipi di errore altrimenti difficili da contenere, tra cui implementazioni del codice non riuscite o richieste compromesse o che attivano una modalità di errore specifica, note anche come richieste poison pill.

Guida all'implementazione

Su una nave gli scomparti permettono di limitare la falla di uno scafo a una sola sezione dello scafo. In sistemi complessi questo modello viene spesso replicato per consentire l'isolamento degli errori. Le limitazioni per l'isolamento degli errori riducono l'effetto di un errore all'interno di un carico di lavoro a un numero limitato di componenti. I componenti al di fuori della barriera non subiscono gli effetti del guasto. Utilizzando più barriere per l'isolamento dei guasti, puoi limitare l'impatto sul carico di lavoro. In AWS i clienti possono usare più zone di disponibilità e regioni per fornire l'isolamento degli errori, ma questo concetto può essere esteso anche all'architettura del carico di lavoro.

Il carico di lavoro complessivo viene partizionato in celle tramite una chiave di partizione. Questa chiave deve essere allineata alla granularità del servizio o al modo naturale in cui il carico di lavoro del servizio può essere suddiviso con interazioni minime tra celle. Esempi di chiavi di partizione sono un ID cliente, un ID risorsa o qualsiasi altro parametro facilmente accessibile nella maggior parte delle chiamate API. Un livello di instradamento alle celle distribuisce le richieste a singole celle in base alla chiave di partizione e presenta un unico endpoint ai client.

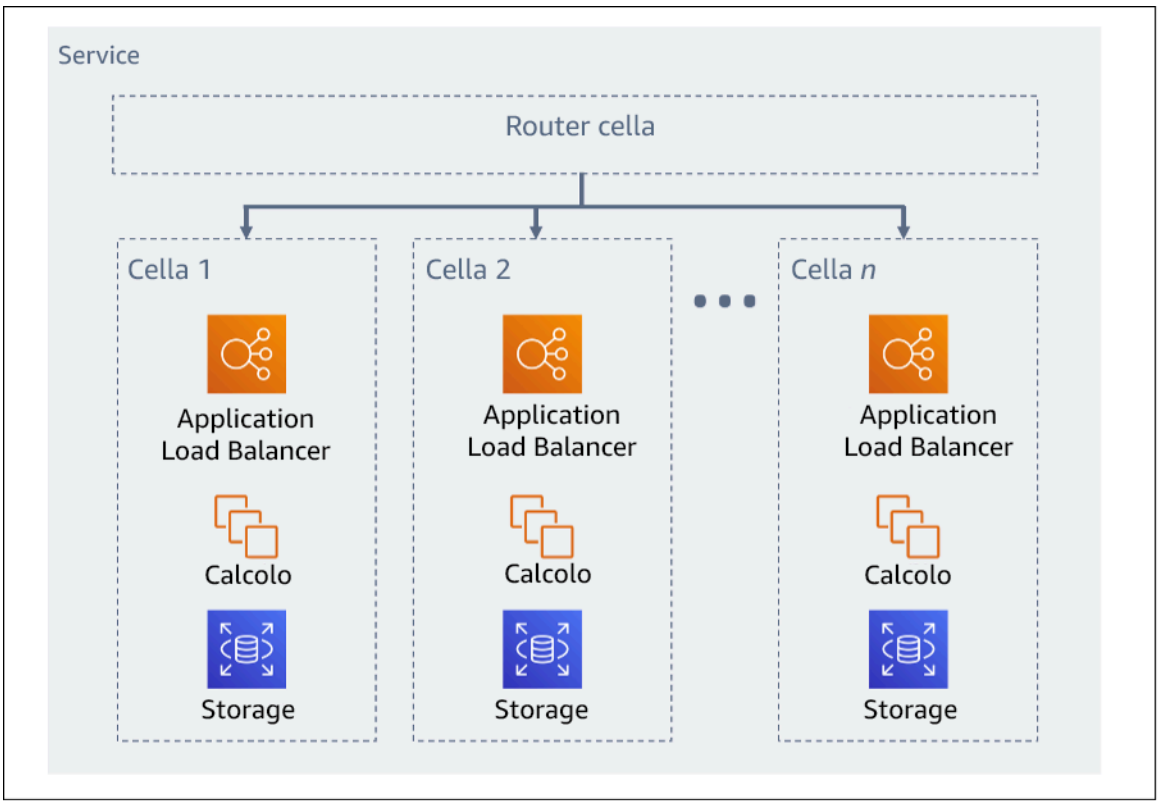


Figura 11. Architettura basata su celle

Passaggi dell'implementazione

Nel progettare un'architettura basata su celle, devi tenere conto di diversi aspetti della progettazione:

1. Chiave di partizione: la scelta della chiave di partizione impone alcune considerazioni speciali.
   - Questa chiave deve essere allineata alla granularità del servizio o al modo naturale in cui il carico di lavoro del servizio può essere suddiviso con interazioni minime tra celle. Alcuni esempi: `ID cliente` oppure `ID risorsa`.
   - La chiave di partizione deve essere disponibile in tutte le richieste, direttamente o in modo da poter essere facilmente dedotta in modo deterministico da altri parametri.
2. Mappatura persistente delle celle: i servizi a monte devono interagire solo con un'unica cella per l'intero ciclo di vita delle risorse correlate.
   - A seconda del carico di lavoro, può essere necessaria una strategia di migrazione delle celle per la migrazione dei dati da una cella a un'altra. Un possibile scenario in cui è necessaria la migrazione delle celle è quando una risorsa o un utente specifico nel carico di lavoro diventa troppo grande e richiede una cella dedicata.
   - Le celle non devono condividere lo stato o i componenti.

- Di conseguenza, l'interazione tra celle deve essere evitata o mantenuta al minimo, in quanto le interazioni creano dipendenze tra le celle e riducono quindi i vantaggi forniti dall'isolamento degli errori.

3. Livello di instradamento: è un componente condiviso tra celle e di conseguenza non può seguire la stessa strategia di compartimentazione applicata alle celle.
   - È consigliabile che il livello di instradamento distribuisca richieste a singole celle usando un algoritmo di mappatura delle partizioni efficiente in termini di risorse di calcolo, ad esempio combinando funzioni hash crittografiche e aritmetica modulare per mappare le chiavi di partizione alle celle.
   - Per evitare l'impatto su più celle, il livello di instradamento deve restare il più semplice e orizzontalmente scalabile possibile, evitando logica di business complessa in questo livello. Questo approccio offre il vantaggio aggiuntivo di semplificare la comprensione del suo comportamento previsto in ogni momento, permettendo test esaustivi. Come spiegato da Colm MacCárthaigh in Reliability, constant work, and a good cup of coffee, progettazioni semplici e modelli di lavoro costanti producono sistemi affidabili e riducono l'antifragilità.
4. Dimensione delle celle: le celle devono avere una dimensione massima che non deve essere superata.
   - La dimensione massima deve essere identificata attraverso l'esecuzione di test completi, fino a raggiungere i punti di rottura e definire i margini operativi. Per ulteriori informazioni su come implementare procedure di test, consulta REL07-BP04 Esecuzione di un test di carico sul carico di lavoro
   - L'aumento del carico di lavoro complessivo deve essere gestito tramite l'aggiunta di celle, in modo da poterlo dimensionare in base al crescere della domanda.
5. Strategie multi-Az e multi-regione: è consigliabile utilizzare più livelli di resilienza a tipi di errore diversi.
   - Per la resilienza, devi utilizzare un approccio che costruisca livelli di difesa. Un livello protegge dalle interruzioni minime e più comuni attraverso la creazione di un'architettura a disponibilità elevata tramite più zone di disponibilità. Un altro livello di difesa è destinato a proteggere da eventi rari come disastri naturali diffusi e interruzioni a livello regionale. Questo secondo livello implica l'architettura dell'applicazione in modo che si estenda su più Regioni AWS. L'implementazione di una strategia multi-regione per il tuo carico di lavoro aiuta a proteggerlo da disastri naturali diffusi che colpiscono un'ampia regione geografica di un paese o da guasti tecnici di portata regionale. Tieni presente che l'implementazione di un'architettura multi-regione può essere molto complessa e di solito non è necessaria per la maggior parte dei carichi di

lavoro. Per ulteriori informazioni, consulta REL10-BP02 Selezione delle posizioni appropriate per la tua implementazione multiposizione.

6. Implementazione del codice: una strategia di implementazione scaglionata del codice è preferibile rispetto all'implementazione di modifiche del codice a tutte le celle contemporaneamente.
   - In questo modo, è possibile ridurre al minimo eventuali errori in più celle a causa di un'implementazione non corretta o dell'errore umano. Per ulteriori informazioni, consulta Automatizzazione di implementazioni pratiche e sicure.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Risorse

Best practice correlate:

- REL07-BP04 Esecuzione di un test di carico sul carico di lavoro
- REL10-BP02 Selezione delle posizioni appropriate per la tua implementazione multiposizione

Documenti correlati:

- Reliability, constant work, and a good cup of coffee
- AWS e compartimentazione
- Isolamento del carico di lavoro tramite sharding casuale
- Automatizzazione di implementazioni pratiche e sicure

Video correlati:

- AWS re:Invent 2018: Cicli chiusi e menti aperte: come assumere il controllo di sistemi grandi e piccoli
- AWS re:Invent 2018: AWS riduce al minimo il raggio di esplosione degli errori (ARC338)
- Partizionamento casuale: AWS re:Invent 2019: Introduzione alla Libreria dei costruttori di Amazon (DOP328)
- AWS Summit ANZ 2021:Gli errori si verificano sempre e ovunque: una progettazione per la resilienza

Esempi correlati:

- Well-Architected Lab: Isolamento degli errori con il partizionamento casuale

REL 11. Come si progetta il carico di lavoro affinché resista ai guasti dei componenti?

I carichi di lavoro con requisiti di disponibilità elevata e MTTR (Mean Time To Recovery) basso devono essere progettati per garantire la resilienza.

Best practice

- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP02 Failover e passaggio a risorse integre
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo durante il ripristino
- REL11-BP05 Utilizzo della stabilità statica per evitare un comportamento bimodale
- REL11-BP06 Invio di notifiche quando gli eventi influiscono sulla disponibilità
- REL11-BP07 Progettazione del prodotto in modo da soddisfare gli obiettivi di disponibilità e i contratti sul livello di servizio per i tempi di attività

REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Monitora costantemente lo stato del carico di lavoro, in modo che tu e i tuoi sistemi automatizzati siate consapevoli di errori o guasti non appena si verificano. Monitora gli indicatori chiave di prestazioni (KPI) in base al valore aziendale.

Tutti i meccanismi di ripristino e correzione devono essere in grado di rilevare rapidamente i problemi. I guasti tecnici devono essere rilevati prima in modo che possano essere risolti. Tuttavia, la disponibilità si basa sulla capacità del carico di lavoro di fornire valore aziendale, quindi gli indicatori chiave di prestazione (KPI) che misurano questo aspetto devono far parte della strategia di rilevamento e correzione.

Risultato desiderato: I componenti essenziali di un carico di lavoro vengono monitorati in modo indipendente per rilevare guasti e fornire avvisi quando e dove si verificano.

Anti-pattern comuni:

- Non sono stati configurati allarmi, pertanto le interruzioni si verificano senza notifica.
- Gli allarmi esistono, ma a soglie che non forniscono tempo adeguato per reagire.

- I parametri non vengono raccolti abbastanza spesso da soddisfare l'obiettivo di tempo di ripristino (RTO, recovery time objective).
- Solo le interfacce del carico di lavoro rivolte al cliente vengono monitorate attivamente.
- Viene effettuata solo la raccolta di parametri tecnici, senza includere quelli delle funzioni aziendali.
- Non è presente alcun parametro che misuri l'esperienza utente del carico di lavoro.
- Vengono creati troppi monitoraggi.

Vantaggi dell'adozione di questa best practice: Eseguire un monitoraggio appropriato a tutti i livelli consente di ridurre i tempi di rilevamento, velocizzando quindi il ripristino.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Identifica tutti i carichi di lavoro che verranno esaminati per il monitoraggio. Dopo aver identificato tutti i componenti del carico di lavoro da monitorare, devi determinare l'intervallo di monitoraggio. L'intervallo di monitoraggio ha un impatto diretto sulla velocità con cui il ripristino viene avviato, che dipende dal tempo impiegato per rilevare un errore. Il tempo medio di rilevamento (MTTD) è il tempo che intercorre tra il verificarsi di un guasto e l'inizio delle operazioni di riparazione. L'elenco dei servizi deve essere ampio e completo.

Il monitoraggio deve includere tutti i livelli dello stack applicativo, come applicazione, piattaforma, infrastruttura e rete.

La strategia di monitoraggio deve tenere in considerazione l'impatto di guasti nell'area grigia. Per ulteriori dettagli sui guasti nell'area grigia, consulta il whitepaper Gray failures in the Advanced Multi-AZ Resilience Patterns

Passaggi dell'implementazione

- L'intervallo di monitoraggio dipende dalla velocità con cui è necessario ripristinare Il tempo di ripristino dipende dal tempo necessario a ripristinare, perciò è necessario determinare la frequenza della raccolta considerando tale tempo e l'obiettivo di tempo di ripristino (RTO, recovery time objective).
- Configura il monitoraggio dettagliato per componenti e servizi gestiti.
  - Determina se il monitoraggio dettagliato per le istanze EC2 e Auto Scaling è necessario. Il monitoraggio dettagliato fornisce metriche a intervalli di un minuto, mentre il monitoraggio predefinito fornisce metriche a intervalli di cinque minuti.

  - Determina se il monitoraggio avanzato per RDS è necessario. Il monitoraggio avanzato utilizza un agente sulle istanze RDS per ottenere informazioni utili su diversi processi o thread.
  - Determina i requisiti di monitoraggio dei componenti serverless critici per Lambda, API Gateway, Amazon EKS, Amazon ECSe tutti i tipi di sistema di bilanciamento del carico.
  - Determina i requisiti di monitoraggio dei componenti di archiviazione per Amazon S3, Amazon FSx, Amazon EFSe Amazon EBS.
- Crea metriche personalizzate per misurare gli indicatori di prestazione (KPI) fondamentali per il tuo business. I carichi di lavoro implementano funzioni aziendali fondamentali, che devono essere utilizzate come KPI che aiutano a identificare quando si verifica un problema indiretto.
- Monitoraggio della presenza di errori nell'esperienza utente tramite le canary degli utenti Test delle transazioni sintetiche (noto anche come test canary, ma da non confondere con l'implementazione canary) è uno dei processi di test più importanti in quanto è in grado di eseguire e simulare il comportamento dei clienti. Esegui questi test costantemente sugli endpoint del carico di lavoro da diverse posizioni remote.
- Crea metriche personalizzate che monitorino l'esperienza dell'utente. Dotare l'esperienza del cliente di strumenti consente di determinare quando essa peggiora.
- Imposta allarmi per rilevare quando una qualsiasi parte del carico di lavoro non funziona correttamente e per indicare quando dimensionare automaticamente le risorse. È possibile mostrare visivamente gli allarmi sulle dashboard, inviarli tramite Amazon SNS o e-mail e utilizzarli con Auto Scaling per aumentare o ridurre le risorse del carico di lavoro.
- Crea dashboard per visualizzare le metriche. Utilizza le dashboard per visualizzare tendenze, valori anomali e altri indicatori di potenziali problemi, oppure per fornire un'indicazione dei problemi che potresti voler approfondire.
- Crea il monitoraggio del tracciamento distribuito per i tuoi servizi. Con il monitoraggio distribuito puoi comprendere le prestazioni della tua applicazione e dei relativi servizi sottostanti per identificare e risolvere la causa ultima di problemi ed errori riguardanti le prestazioni.
- Utilizza CloudWatch oppure X-Rayper creare dashboard di sistemi di monitoraggio e di raccolta dati in una regione e in un account separati.
- Crea l'integrazione per Amazon Health Aware per consentire il monitoraggio della visibilità sulle risorse AWS che potrebbero presentare un deterioramento. Per i carichi di lavoro aziendali essenziali, questa soluzione fornisce l'accesso ad avvisi proattivi e in tempo reale per i servizi AWS.

Risorse

Best practice correlate:

- Definizione di disponibilità
- REL11-BP06 Invio di notifiche quando gli eventi influiscono sulla disponibilità

Documenti correlati:

- Amazon CloudWatch Synthetics consente di creare i Canary dell'utente
- Abilitare o disabilitare il monitoraggio dettagliato della propria istanza
- Monitoraggio avanzato
- Monitoring Your Auto Scaling Groups and Instances Using Amazon CloudWatch
- Pubblicazione di parametri personalizzati
- Utilizzo degli allarmi di Amazon CloudWatch
- Using CloudWatch Dashboards
- Using Cross Region Cross Account CloudWatch Dashboards
- Using Cross Region Cross Account X-Ray Tracing
- Understanding availability
- Implementing Amazon Health Aware (AHA)

Video correlati:

- Mitigating gray failures

Esempi correlati:

- Well-Architected Lab: Level 300: Implementing Health Checks and Managing Dependencies to Improve Reliability
- One Observability Workshop: Explore X-Ray

Strumenti correlati:

- CloudWatch
- CloudWatch X-Ray

REL11-BP02 Failover e passaggio a risorse integre

Se si verifica un errore in una risorsa, le risorse integre dovrebbero continuare a soddisfare le richieste. Per posizioni compromesse (ad esempio una zona di disponibilità o una Regione AWS), assicurati di disporre di sistemi che possano eseguire il failover e passare a risorse integre in posizioni non danneggiate.

Durante la progettazione di un servizio, distribuisci il carico tra risorse, zone di disponibilità o regioni. Pertanto, il guasto o la compromissione di una singola risorsa può essere mitigato spostando il traffico sulle risorse integre rimanenti. Considera come vengono rilevati e indirizzati i servizi in caso di guasto.

Progetta i tuoi servizi tenendo a mente il recupero dai guasti. In AWS, progettiamo servizi per ridurre al minimo i tempi di recupero da guasti e l'impatto sui dati. I nostri servizi utilizzano principalmente archivi di dati che riconoscono le richieste solo dopo che queste sono state archiviate in modo duraturo su più repliche in una Regione. Sono costruiti con il criterio dell'isolamento basato sulle celle ed utilizzano l'isolamento dei guasti fornito dalle zone di disponibilità. Facciamo ampio uso dell'automazione nelle nostre procedure operative. Ottimizziamo anche la nostra funzionalità di sostituzione e riavvio per un ripristino rapidamente dalle interruzioni.

I modelli e i progetti che consentono il failover variano a seconda dei servizi della AWS. Molti servizi AWS gestiti nativi si trovano in più zone di disponibilità (come Lambda oAPI Gateway) in modo nativo. Altri servizi AWS (come EC2 ed EKS) richiedono procedure ottimali specifiche per supportare il failover delle risorse o l'archiviazione di dati tra le zone di disponibilità.

Il monitoraggio deve essere impostato per verificare che la risorsa di failover sia integra, tenere traccia dell'avanzamento del failover delle risorse e monitorare il ripristino dei processi aziendali.

Risultato desiderato: I sistemi sono in grado di utilizzare automaticamente o manualmente nuove risorse per il ripristino dopo un evento di deterioramento.

Anti-pattern comuni:

- La pianificazione degli errori non fa parte della fase di pianificazione e progettazione.
- L'obiettivo del tempo di ripristino (RTO) e l'obiettivo del punto di ripristino (RPO) non sono stabiliti.
- Monitoraggio insufficiente per rilevare risorse difettose.
- Isolamento adeguato dei domini di errore.
- Il failover multi-regione non è considerato.
- Il rilevamento dei guasti è troppo sensibile o aggressivo quando si decide di eseguire il failover.

- Non è possibile testare o convalidare il progetto di failover.
- Esecuzione dell'automazione del risanamento automatico, ma senza la notifica della necessità di una correzione.
- Mancanza di un periodo di mitigazione per evitare che l'errore si ripresenti troppo presto.

Vantaggi dell'adozione di questa best practice: È possibile creare sistemi più resilienti che mantengano l'affidabilità in caso di guasti eseguendo prima un deterioramento lento e poi un ripristino rapido.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

I servizi AWS, come Elastic Load Balancing e Amazon EC2 Auto Scaling, aiutano a distribuire il carico tra risorse e zone di disponibilità. Pertanto, il guasto di una singola risorsa (come un'istanza EC2) o la compromissione di una zona di disponibilità possono essere mitigati spostando il traffico sulle risorse integre rimanenti.

Per i carichi di lavoro multi-regione, i progetti sono più complicati. Ad esempio, le repliche di lettura multi-regione consentono di implementare i dati su Regioni AWS multiple. Tuttavia, il failover è ancora necessario per promuovere la replica di lettura a principale e quindi indirizzare il traffico verso il nuovo endpoint. Amazon Route 53, Route 53 Route 53 ARC, CloudFront e AWS Global Accelerator possono aiutare a instradare il traffico tra le Regioni AWS.

Servizi AWS come Amazon S3, Lambda, API Gateway, Amazon SQS, Amazon SNS, Amazon SES, Amazon Pinpoint, Amazon ECR, AWS Certificate Manager, EventBridge o Amazon DynamoDB vengono implementati automaticamente in più zone di disponibilità da AWS. In caso di guasto, questi servizi AWS instradano automaticamente il traffico verso posizioni integre. I dati sono archiviati in modo ridondante in più zone di disponibilità e rimangono disponibili.

Per Amazon RDS, Amazon Aurora, Amazon Redshift, Amazon EKS o Amazon ECS, l'implementazione Multi-AZ è un'opzione di configurazione. AWS può indirizzare il traffico verso l'istanza integra se viene avviato il failover. Questa azione di failover può essere intrapresa direttamente da AWS o su richiesta del cliente.

Per istanze Amazon EC2, Amazon Redshift, attività Amazon ECS o pod Amazon EKS, sei tu a scegliere in quali zone di disponibilità eseguire la distribuzione. Per alcuni progetti, Elastic Load Balancing fornisce la soluzione per rilevare istanze in zone corrotte e instradare il traffico verso quelle

integre. Elastic Load Balancing può anche indirizzare il traffico verso i componenti del data center on-premise.

Per il failover del traffico multi-regione, il reindirizzamento può sfruttare Amazon Route 53, Route 53 ARC, AWS Global Accelerator, Route 53 Private DNS for VPCs o CloudFront per fornire una modalità per definire domini Internet e assegnare policy di routing, compresi i controlli dell'integrità, per instradare il traffico verso regioni integre. AWS Global Accelerator fornisce indirizzi IP statici che operano come punto di ingresso fisso all'applicazione, che indirizzano il traffico verso gli endpoint delle Regioni AWS di tua scelta utilizzando la rete AWS globale anziché Internet per prestazioni e affidabilità migliori.

Passaggi dell'implementazione

- Crea progetti di failover per tutte le applicazioni e i servizi appropriati. Isola ogni componente dell'architettura e crea progetti di failover che soddisfino l'RTO e l'RPO per ogni componente.
- Configura ambienti inferiori (come sviluppo o test) con tutti i servizi necessari per disporre di un piano di failover. Implementa le soluzioni utilizzando l'infrastruttura come codice (IaC) per garantire la ripetibilità.
- Configura un sito di ripristino, ad esempio una seconda regione, per implementare e testare i progetti di failover. Se necessario, le risorse per i test possono essere configurate temporaneamente per limitare i costi aggiuntivi.
- Determina quali piani di failover sono automatizzati da AWS, quali possono essere automatizzati da un processo DevOps e quali possono essere manuali. Documenta e misura l'RTO e l'RPO di ogni servizio.
- Crea un playbook per il failover e includi tutti i passaggi necessari per eseguire il failover di ogni risorsa, applicazione e servizio.
- Crea un playbook di failback e includi tutti i passaggi per eseguire il failback (con tempistiche) di ogni risorsa, applicazione e servizio.
- Crea un piano per avviare e testare il playbook. Usa simulazioni e test del caos per testare i passaggi e l'automazione del playbook.
- Per posizioni compromesse (ad esempio una zona di disponibilità o una Regione AWS), assicurati di disporre di sistemi che possano eseguire il failover e passare a risorse integre in posizioni non danneggiate. Verifica la quota, i livelli di dimensionamento automatico e le risorse in esecuzione prima dei test di failover.

Risorse

Best practice Well-Architected correlate:

- REL13- Pianificazione per il disaster recovery (DR)
- REL10 - Utilizzo dell'isolamento dei guasti per proteggere il carico di lavoro

Documenti correlati:

- Setting RTO and RPO Targets
- Set up Route 53 ARC with application loadbalancers
- Failover using Route 53 Weighted routing
- DR with Route 53 ARC
- EC2 with autoscaling
- EC2 Deployments - Multi-AZ
- ECS Deployments - Multi-AZ
- Switch traffic using Route 53 ARC
- Lambda with an Application Load Balancer and Failover
- ACM Replication and Failover
- Parameter Store Replication and Failover
- ECR cross region replication and Failover
- Secrets manager cross region replication configuration
- Enable cross region replication for EFS and Failover
- EFS Cross Region Replication and Failover
- Networking Failover
- S3 Endpoint failover using MRAP
- Crea una replica tra regioni per S3
- Failover Regional API Gateway with Route 53 ARC
- Failover using multi-region global accelerator
- Failover with DRS
- Creating Disaster Recovery Mechanisms Using Amazon Route 53

Esempi correlati:

- Disaster Recovery on AWS
- Elastic Disaster Recovery on AWS

REL11-BP03 Automatizzazione della riparazione a tutti i livelli

Al rilevamento di un guasto, utilizza funzionalità automatizzate per eseguire azioni da correggere. I guasti possono essere riparati automaticamente tramite meccanismi di servizio interni oppure riavviando o rimuovendo le risorse tramite azioni correttive.

Per applicazioni gestite dal cliente e per il ripristino tra regioni, è possibile attingere a modelli di ripristino e processi di riparazione automatizzati dalle best practice esistenti.

La possibilità di riavviare o rimuovere una risorsa è uno strumento importante per risolvere i guasti. Una best practice consiste nel rendere i servizi stateless, ove possibile. In questo modo si evita la perdita di dati o di disponibilità durante il riavvio della risorsa. Nel cloud è possibile, e in genere si dovrebbe, sostituire l'intera risorsa (ad esempio, un'istanza di calcolo o una funzione serverless) come parte del riavvio. Il riavvio stesso è un modo semplice e affidabile per eseguire il ripristino in caso di guasto. Molti tipi diversi di guasto si verificano nei carichi di lavoro. Possono verificarsi guasti a livello di hardware, software, comunicazione e operazioni.

Il riavvio o i nuovi tentativi come pratiche risolutive si applicano anche alle richieste di rete. Adotta lo stesso approccio di ripristino sia a un timeout di rete sia a un guasto di dipendenza in cui la dipendenza restituisce un guasto. Entrambi gli eventi hanno un effetto simile sul sistema, quindi piuttosto che tentare di trasformare entrambi gli eventi in un caso speciale, adotta una strategia analoga di nuovi tentativi limitati con un backoff e un jitter esponenziali. La capacità di riavvio è un meccanismo di ripristino presente nelle architetture di cluster ROC (Recovery-oriented computing) e ad alta disponibilità.

Risultato desiderato: Vengono eseguite azioni automatiche di risoluzione a seguito del rilevamento di un errore.

Anti-pattern comuni:

- Provisioning di risorse senza dimensionamento automatico.
- Implementazione individuale di applicazioni in istanze/container.
- Distribuzione di applicazioni che non possono essere distribuite in più posizioni senza utilizzare il ripristino automatico.

- Riparazione manuale delle applicazioni che il dimensionamento e il ripristino automatici non sono stati in grado di riparare.
- Nessuna automazione dei database di failover.
- Mancanza di metodi automatizzati per reinstradare il traffico verso nuovi endpoint.
- Nessuna replica dell'archiviazione.

Vantaggi dell'adozione di questa best practice: La riparazione automatica può ridurre il tempo medio di ripristino e migliorare la disponibilità.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

I progetti per Amazon EKS o altri servizi Kubernetes devono includere il numero minimo e massimo di repliche o di stateful set e la dimensione minima dei cluster e dei gruppi di nodi. Questi meccanismi forniscono una quantità minima di risorse di elaborazione continuamente disponibili mentre riparano automaticamente eventuali guasti utilizzando il piano di controllo Kubernetes.

I modelli di progettazione a cui si accede tramite un sistema di bilanciamento del carico che utilizza cluster di calcolo dovrebbero sfruttare i gruppi Auto Scaling. Elastic Load Balancing (ELB) distribuisce automaticamente il traffico delle applicazioni in entrata su più destinazioni e applicazioni virtuali in una o più zone di disponibilità (AZ).

I progetti basati su cluster computing che non utilizzano il bilanciamento del carico devono avere dimensioni progettate per la perdita di almeno un nodo. Ciò consentirà al servizio di rimanere in esecuzione con una capacità potenzialmente ridotta durante il ripristino di un nuovo nodo. Servizi di esempio sono Mongo, DynamoDB Accelerator, Amazon Redshift, Amazon EMR, Cassandra, Kafka, MSK-EC2, Couchbase, ELK e Amazon OpenSearch Service. Molti di questi servizi possono essere progettati con funzionalità di riparazione automatica aggiuntive. Alcune tecnologie di cluster devono generare un avviso in caso di perdita di un nodo attivando un flusso di lavoro automatico o manuale per creare un nuovo nodo. È possibile automatizzare questo flusso di lavoro utilizzando AWS Systems Manager per risolvere rapidamente i problemi.

Amazon EventBridge può essere utilizzato per monitorare e filtrare eventi come allarmi CloudWatch o cambiamenti di stato in altri servizi AWS. In base alle informazioni sugli eventi, può quindi richiamare AWS Lambda, Systems Manager Automation o altre destinazioni per eseguire una logica di riparazione personalizzata sul carico di lavoro. È possibile configurare Amazon EC2 Auto Scaling per verificare lo stato delle istanze EC2. Se l'istanza è in uno stato diverso da quello in esecuzione o se

lo stato del sistema è danneggiato, Amazon EC2 Auto Scaling considera l'istanza come non integra e ne avvia una sostitutiva. Per le sostituzioni su larga scala (ad esempio la perdita di un'intera zona di disponibilità), è preferibile adottare la stabilità statica per ottenere un'elevata disponibilità.

Passaggi dell'implementazione

- Utilizza gruppi di Auto Scaling per distribuire livelli in un carico di lavoro. Auto Scaling è in grado di eseguire il risanamento automatico sulle applicazioni stateless e aggiungere o rimuovere capacità.
- Per le istanze di calcolo indicate in precedenza, usa il bilanciamento del carico e scegli il tipo di sistema di bilanciamento del carico appropriato.
- Considera l'opzione della riparazione per Amazon RDS. Con le istanze di standby, configura il failover automatico verso l'istanza di standby. Per le repliche in lettura Amazon RDS, è necessario un flusso di lavoro automatizzato per rendere primaria una replica di lettura.
- Implementa il ripristino automatico su istanze EC2 che includono applicazioni distribuite non implementabili in più sedi e che possono tollerare il riavvio in caso di guasto. Il ripristino automatico può essere utilizzato per sostituire l'hardware guasto e riavviare l'istanza quando l'applicazione non è in grado di essere distribuita in più posizioni. I metadati dell'istanza e gli indirizzi IP associati vengono conservati, così come i volumi EBS e i punti di montaggio su Amazon Elastic File System o i file system per Lustre e Windows. Utilizzando AWS OpsWorkspuoi configurare la riparazione automatica delle istanze EC2 a livello del layer.
- Implementa il ripristino automatico utilizzando AWS Step Functions e AWS Lambda quando non è possibile utilizzare il dimensionamento automatico o il ripristino automatico o quando il ripristino automatico non va a buon fine. Quando non puoi utilizzare il dimensionamento automatico né il ripristino automatico o il ripristino automatico non riesce, puoi automatizzare la riparazione utilizzando AWS Step Functions e AWS Lambda.
- Amazon EventBridge può essere usato per monitorare e filtrare eventi come avvisi CloudWatch o cambiamenti di stato in altri servizi AWS. In base alle informazioni sugli eventi, può quindi richiamare AWS Lambda (o altre destinazioni) per eseguire una logica di riparazione personalizzata sul tuo carico di lavoro.

Risorse

Best practice correlate:

- Definizione di disponibilità
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Documenti correlati:

- Come funziona AWS Auto Scaling
- Ripristino automatico Amazon EC2
- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- What is Amazon FSx for Lustre?
- What is Amazon FSx for Windows File Server?
- AWS OpsWorks: utilizzare il ripristino automatico per sostituire le istanze in errore
- Che cos'è AWS Step Functions?
- Che cos'è AWS Lambda?
- Che cos'è Amazon EventBridge?
- Utilizzo degli allarmi di Amazon CloudWatch
- Amazon RDS Failover
- SSM - Systems Manager Automation
- Resilient Architecture Best Practices

Video correlati:

- Automatically Provision and Scale OpenSearch Service
- Amazon RDS Failover Automatically

Esempi correlati:

- Workshop su Auto Scaling
- Workshop su Amazon RDS Failover

Strumenti correlati:

- CloudWatch
- CloudWatch X-Ray

REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo durante il ripristino

I piani di controllo forniscono le API amministrative utilizzate per creare, leggere e descrivere, aggiornare, eliminare ed elencare (CRUDL) risorse, mentre i piani di dati gestiscono il traffico quotidiano del servizio. Durante l'implementazione di risposte di ripristino o mitigazione a eventi che possono influire sulla resilienza, concentrati sull'utilizzo di un numero minimo di operazioni del piano di controllo per ripristinare, ridimensionare, ristabilire, riparare il servizio o eseguirne il failover. Le operazioni del piano dati dovrebbero avere la precedenza su qualsiasi attività durante questi eventi che causano deterioramento.

Ad esempio, le seguenti sono tutte azioni del piano di controllo: avvio di una nuova istanza di calcolo, creazione di storage a blocchi e descrizione dei servizi di coda. Quando avvii istanze di calcolo, il piano di controllo deve eseguire diverse attività, come trovare un host fisico con capacità, allocare interfacce di rete, preparare volumi di storage a blocchi locali, generare credenziali e aggiungere regole di sicurezza. I piani di controllo tendono ad avere un'orchestrazione complicata.

Risultato desiderato: Quando lo stato di risorsa viene compromesso, il sistema è in grado di ripristinarsi automaticamente o manualmente spostando il traffico da risorse danneggiate a risorse integre.

Anti-pattern comuni:

- Dipendenza dalla modifica dei record DNS per reindirizzare il traffico.
- Dipendenza dalle operazioni di dimensionamento del piano di controllo per sostituire i componenti danneggiati a causa di un provisioning delle risorse insufficiente.
- Affidarsi ad azioni intense, multiservizio e multi-API del piano di controllo per porre rimedio a qualsiasi categoria di deterioramento.

Vantaggi dell'adozione di questa best practice: Una maggiore percentuale di successo in termini di riparazione automatica può ridurre il tempo medio di ripristino e migliorare la disponibilità del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: Medio: per determinati tipi di deterioramento del servizio, vengono compromessi i piani di controllo. Le dipendenze dall'uso intenso del piano di controllo per la riparazione possono aumentare il tempo di ripristino (RTO) e il tempo medio di ripristino (MTTR).

Guida all'implementazione

Per limitare le azioni del piano dati, esegui una valutazione servizio per servizio per determinare le azioni necessarie per ripristinarlo.

Sfrutta Amazon Route 53 Application Recovery Controller per spostare il traffico DNS. Queste funzionalità monitorano continuamente la capacità dell'applicazione di ristabilirsi dai guasti e consentono di controllarne il ripristino su più Regioni AWS, zone di disponibilità e on-premise.

Le policy di instradamento di Route 53 utilizzano il piano di controllo, quindi non fare affidamento su di esso per il ripristino. I piani dati di Route 53 rispondono alle query DNS ed eseguono e valutano i controlli di integrità. Sono distribuiti a livello globale e progettati per un accordo sul livello di servizio (SLA) con disponibilità al 100%.

Le API e la console di gestione di Route 53, dove si creano, aggiornano ed eliminano le risorse di Route 53, funzionano su piani di controllo progettati per privilegiare la forte coerenza e la durata necessarie per la gestione del DNS. A tal fine, i piani di controllo sono situati in un'unica regione: Stati Uniti orientali (Virginia settentrionale). Sebbene entrambi i sistemi siano costruiti per essere molto affidabili, i piani di controllo non sono inclusi nello SLA. Possono verificarsi eventi rari in cui la progettazione resiliente del piano dati consente di mantenere la disponibilità mentre i piani di controllo non lo fanno. Per i meccanismi di ripristino di emergenza e failover, utilizzare le funzioni del piano dati per garantire la migliore affidabilità possibile.

Per Amazon EC2, utilizzare progetti di stabilità statica per limitare le azioni del piano di controllo. Le azioni del piano di controllo includono l'aumento delle risorse, in maniera individuale o utilizzando gruppi Auto Scaling (ASG). Per ottenere i massimi livelli di resilienza, è necessario fornire una capacità sufficiente nel cluster utilizzato per il failover. Se è necessario limitare questa soglia di capacità, imposta acceleratori sull'intero sistema end-to-end per limitare in modo sicuro il traffico totale che raggiunge il set limitato di risorse.

L'utilizzo di servizi come Amazon DynamoDB, Amazon API Gateway, sistemi di bilanciamento del carico e AWS Lambda serverless avviene sfruttando il piano dati. Tuttavia, la creazione di nuove funzioni, sistemi di bilanciamento del carico, gateway API o tabelle DynamoDB è un'azione del piano di controllo e deve essere completata prima del deterioramento come preparazione a un evento e test delle azioni di failover. Per Amazon RDS, le azioni del piano dati consentono l'accesso ai dati.

Per ulteriori informazioni sui piani dati, sui piani di controllo e su come AWS costruisce i servizi per soddisfare gli obiettivi di alta disponibilità, consulta Stabilità statica utilizzando le zone di disponibilità.

Capire quali operazioni sono sul piano dati e quali sul piano di controllo.

Passaggi dell'implementazione

Per ogni carico di lavoro che deve essere ripristinato dopo un evento di deterioramento, valuta il runbook di failover, il design ad alta disponibilità, il progetto di riparazione automatica o il piano di ripristino delle risorse HA. Identifica ogni azione che potrebbe essere considerata un'azione del piano di controllo.

Prendi in considerazione la possibilità di modificare l'azione di controllo in un'azione del piano dati:

- Auto Scaling (piano di controllo) rispetto alle risorse Amazon EC2 predimensionate (piano dati)
- Esegui la migrazione verso Lambda e i relativi metodi di dimensionamento (piano dati) oppure verso Amazon EC2 e ASG (piano di controllo)
- Valuta qualsiasi progetto utilizzando Kubernetes e considerando la natura delle azioni del piano di controllo. L'aggiunta di pod è un'azione del piano dati in Kubernetes. Le azioni devono limitarsi all'aggiunta di pod e non all'aggiunta di nodi. L'utilizzo di nodi con provisioning eccessivo è il metodo preferibile per limitare le azioni del piano di controllo

Prendi in considerazione approcci alternativi che consentano alle azioni del piano dati di incidere sulla stessa correzione.

- Modifica del record di Route 53 (piano di controllo) o Route 53 ARC (piano dati)
- Controlli dell'integrità di Route 53 per aggiornamenti più automatizzati

Se il servizio è mission critical, prendi in considerazione alcuni servizi in una regione secondaria per consentire più azioni del piano di controllo e del piano dati in una regione non interessata dal problema.

- Amazon EC2 Auto Scaling o Amazon EKS in una regione primaria rispetto a Amazon EC2 Auto Scaling o Amazon EKS in una regione secondaria e instradamento del traffico verso una regione secondaria (azione del piano di controllo)
- Crea una replica di lettura nella regione secondaria o tenta la stessa azione nella regione principale (azione del piano di controllo)

Risorse

Best practice correlate:

- Definizione di disponibilità
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Documenti correlati:

- Partner APN: partner che possono essere d'aiuto con l'automazione della tua tolleranza ai guasti
- Marketplace AWS: prodotti utilizzabili per la tolleranza ai guasti
- The Amazon Builders' Library: Avoiding overload in distributed systems by putting the smaller service in control
- API Amazon DynamoDB (piano di controllo e piano dati)
- AWS Lambda Executions (suddivise in piano di controllo e piano dati)
- Piano dati di AWS Elemental MediaStore
- Building highly resilient applications using Amazon Route 53 Application Recovery Controller, Part 1: Single-Region stack
- Building highly resilient applications using Amazon Route 53 Application Recovery Controller, Part 2: Multi-Region stack
- Creating Disaster Recovery Mechanisms Using Amazon Route 53
- Che cos'è il Sistema di controllo Route 53 per il ripristino di applicazioni
- Piano di controllo e piano dati di Kubernetes

Video correlati:

- Back to Basics - Using Static Stability
- Building resilient multi-site workloads using AWS global services

Esempi correlati:

- Introducing Amazon Route 53 Application Recovery Controller
- The Amazon Builders' Library: Avoiding overload in distributed systems by putting the smaller service in control
- Building highly resilient applications using Amazon Route 53 Application Recovery Controller, Part 1: Single-Region stack

- Building highly resilient applications using Amazon Route 53 Application Recovery Controller, Part 2: Multi-Region stack
- Stabilità statica utilizzando le zone di disponibilità

Strumenti correlati:

- Amazon CloudWatch
- AWS X-Ray

REL11-BP05 Utilizzo della stabilità statica per evitare un comportamento bimodale

I carichi di lavoro devono essere staticamente stabili e funzionare in una singola modalità normale. Il comportamento bimodale si verifica quando il carico di lavoro presenta un comportamento diverso in modalità normale e in modalità di guasto.

Ad esempio, ciò potrebbe accadere nel momento in cui si prova a ripristinare un guasto nella zona di disponibilità avviando nuove istanze in una zona di disponibilità diversa. Questo approccio può comportare una risposta bimodale durante una modalità di guasto. È invece necessario creare carichi di lavoro che siano staticamente stabili e operino in una sola modalità. In questo esempio, le nuove istanze avrebbero dovuto essere rese disponibili nella seconda zona di disponibilità già prima del guasto. Questo design staticamente stabile verifica che il carico di lavoro funzioni in una sola modalità.

Risultato desiderato: i carichi di lavoro non presentano un comportamento bimodale in modalità normale e in modalità di guasto.

Anti-pattern comuni:

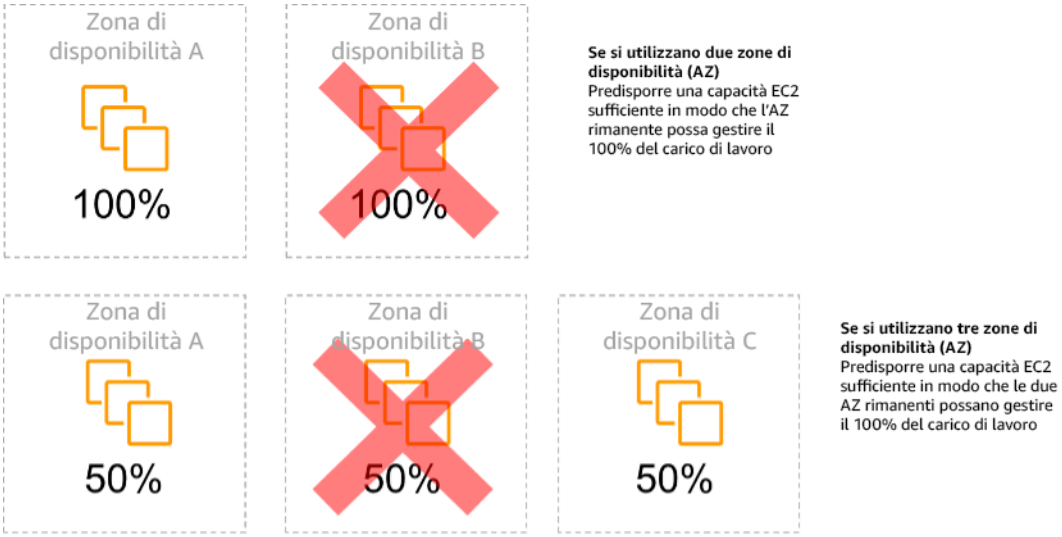
- Supporre che le risorse possano sempre essere rese disponibili indipendentemente dall'ambito del guasto.
- Tentare di acquisire risorse in modo dinamico durante un guasto.
- Non rendere disponibili risorse adeguate tra zone o regioni diverse fino a quando non si verifica un guasto.
- Considerare i progetti staticamente stabili solo per risorse di calcolo.

Vantaggi dell'adozione di questa best practice: I carichi di lavoro eseguiti con progetti staticamente stabili sono in grado di avere risultati prevedibili durante eventi normali e di guasto.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Il comportamento bimodale ha luogo quando il carico di lavoro mostra un comportamento diverso in modalità normale e di guasto, ad esempio facendo affidamento sull'avvio di nuove istanze se una zona di disponibilità presenta un malfunzionamento. Un esempio di comportamento bimodale è quello che si verifica quando design stabili di Amazon EC2 rendono disponibili un numero sufficiente di istanze in ciascuna zona di disponibilità per gestire il carico di lavoro in caso di rimozione di una di tali zone. Elastic Load Balancing o Amazon Route 53 possono effettuare un controllo di integrità per spostare un carico lontano dalle istanze danneggiate. Dopo il trasferimento del traffico, è possibile utilizzare AWS Auto Scaling per sostituire in modo asincrono le istanze della zona interessata dal guasto avviandole nelle zone integre. La stabilità statica per il deployment delle risorse di calcolo (ad esempio istanze EC2 o container) determinerà la massima affidabilità.



Stabilità statica delle istanze EC2 nelle diverse zone di disponibilità

Questo approccio deve essere valutato rispetto al costo associato al modello e al valore aziendale attribuito al mantenimento della disponibilità del carico di lavoro in tutti i casi di resilienza. Fornire una minore capacità di elaborazione e affidarsi all'avvio di nuove istanze in caso di guasto è meno costoso. Tuttavia, in caso di guasti su larga scala, come una zona di disponibilità o un problema a livello regionale, tale approccio è meno efficace, perché si basa su un piano operativo e sulla disponibilità di risorse sufficienti nelle zone o nelle regioni non interessate dal problema.

La soluzione deve inoltre valutare l'affidabilità rispetto ai costi necessari per il carico di lavoro. Gli approcci che garantiscono la stabilità statica si applicano a una varietà di architetture, tra cui istanze

di calcolo distribuite tra zone di disponibilità, progetti di repliche di lettura di database, progetti di cluster Kubernetes (Amazon EKS) e architetture di failover multiregione.

È anche possibile implementare un progetto staticamente più stabile utilizzando più risorse in ciascuna zona. Aggiungendo più zone, si riduce la quantità di elaborazione aggiuntiva necessaria per la stabilità statica.

Un altro esempio di comportamento bimodale potrebbe derivare da un timeout di rete in grado di causare un tentativo di aggiornamento dello stato di configurazione dell'intero sistema. Ciò potrebbe aggiungere un carico imprevisto su un altro componente che potrebbe quindi generare un errore, innescando ulteriori conseguenze impreviste. Questo loop di feedback negativo influisce sulla disponibilità del carico di lavoro. Al contrario, è possibile creare sistemi che siano staticamente stabili e funzionino in una sola modalità. Un progetto staticamente stabile potrebbe eseguire con continuità un'attività e aggiornare sempre, con cadenza regolare, lo stato della configurazione. Quando una chiamata fallisce, il carico di lavoro può utilizzare il valore precedentemente memorizzato nella cache e segnalare un allarme.

Un altro esempio di comportamento bimodale è consentire ai client di bypassare la cache del carico di lavoro quando si verificano dei guasti. Potrebbe sembrare una soluzione che soddisfa le esigenze del client, ma non dovrebbe essere consentita perché modifica in modo significativo le richieste sul carico di lavoro e potrebbe causare dei guasti.

Valuta i carichi di lavoro critici per determinare quali carichi di lavoro richiedono questo tipo di progettazione di resilienza. Per quelli considerati critici, deve essere esaminato ogni componente dell'applicazione. Alcuni tipi di servizi che richiedono valutazioni di stabilità statica sono:

- Calcolo: Amazon EC2, EKS-EC2, ECS-EC2, EMR-EC2
- Database: Amazon Redshift, Amazon RDS, Amazon Aurora
- Storage: Amazon S3 (Zona singola), Amazon EFS (supporti), Amazon FSx (supporti)
- Sistemi di bilanciamento del carico: In base a determinati modelli

Passaggi dell'implementazione

- Realizzare sistemi che siano staticamente stabili e operino in una sola modalità. In questo caso, effettuare il provisioning di un numero sufficiente di istanze in ogni zona o regione di disponibilità per gestire la capacità del carico di lavoro qualora venga rimossa una zona o regione di disponibilità. Per l'indirizzamento verso risorse integre è possibile utilizzare una varietà di servizi, come:

  - Cross Region DNS Routinge
  - Routing Amazon S3 multiregionale MRAP
  - AWS Global Accelerator
  - Amazon Route 53 Application Recovery Controller
- Configura repliche di lettura del database per tenere conto della perdita di una singola istanza primaria o di una replica di lettura. Se il traffico viene servito da repliche di lettura, la quantità in ogni zona di disponibilità e in ogni regione deve corrispondere al fabbisogno complessivo in caso di guasto della zona o della regione.
- Configurare i dati critici nel sistema di archiviazione Amazon S3 progettato per essere staticamente stabile rispetto ai dati archiviati in caso di guasto della zona di disponibilità. Se si verifica un Se viene utilizzata la classe di archiviazione Amazon S3 One Zone-IA questa non deve essere considerata staticamente stabile, poiché la perdita di tale zona riduce al minimo l'accesso ai dati archiviati.
- I sistemi di bilanciamento del carico sono a volte configurati in modo errato o sono progettati per servire una zona di disponibilità specifica. In questo caso, il progetto staticamente stabile potrebbe consistere nel distribuire un carico di lavoro su più zone di disponibilità seguendo un design più complesso. Il design originale potrebbe essere utilizzato per ridurre il traffico tra zone per motivi di sicurezza, latenza o costi.

Risorse

Best practice Well-Architected correlate:

- Definizione di disponibilità
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo durante il ripristino

Documenti correlati:

- Minimizing Dependencies in a Disaster Recovery Plan
- The Amazon Builders' Library: Stabilità statica con le zone di disponibilità
- Fault Isolation Boundaries
- Stabilità statica utilizzando le zone di disponibilità
- Multi-Zone RDS

- Minimizing Dependencies in a Disaster Recovery Plan
- Cross Region DNS Routinge
- Routing Amazon S3 multiregionale MRAP
- AWS Global Accelerator
- Route 53 ARC
- Amazon S3 a singola zona
- Cross Zone Load Balancing

Video correlati:

- Static stability in AWS: AWS re:Invent 2019: Introducing The Amazon Builders' Library (DOP328)

Esempi correlati:

- The Amazon Builders' Library: Stabilità statica con le zone di disponibilità

REL11-BP06 Invio di notifiche quando gli eventi influiscono sulla disponibilità

Le notifiche vengono inviate al rilevamento del superamento delle soglie, anche se l'evento causato dal problema è stato risolto automaticamente.

Il ripristino automatizzato consente al carico di lavoro di risultare affidabile. Tuttavia, potrebbe anche nascondere problemi sottostanti che devono essere risolti. Implementa il monitoraggio e gli eventi appropriati in modo da poter rilevare i modelli di problemi, inclusi quelli risolti dalla diagnostica automatica e risolvere così i problemi della causa principale.

I sistemi resilienti sono progettati in modo che gli eventi di degrado vengano immediatamente comunicati ai team appropriati. Queste notifiche devono essere inviate tramite uno o più canali di comunicazione.

Risultato desiderato: Gli avvisi vengono inviati immediatamente ai team operativi quando vengono superate soglie come i tassi di errore, la latenza o altri parametri critici degli indicatori chiave di prestazione (KPI), in modo che questi problemi vengano risolti il prima possibile e l'impatto sugli utenti sia evitato o ridotto al minimo.

Anti-pattern comuni:

- Invio di un numero eccessivo di avvisi.

- Invio di avvisi non utilizzabili.
- Impostazione di soglie di allarme troppo alte (troppo sensibili) o troppo basse (troppo poco sensibili).
- Mancato invio di avvisi per dipendenze esterne.
- Non considerando guasti nell'area grigia nella progettazione di sistemi di monitoraggio e allarmi.
- Eseguire l'automazione del risanamento, ma senza avvisare il team competente che era necessario un intervento di ripristino.

Vantaggi dell'adozione di questa best practice: Le notifiche di ripristino rendono i team operativi e aziendali consapevoli dei peggioramenti del servizio in modo che possano reagire immediatamente per ridurre al minimo sia il tempo medio di rilevamento (MTTD) che il tempo medio di riparazione (MTTR). Le notifiche degli eventi di ripristino consentono anche di non ignorare i problemi che si verificano di rado.

Livello di rischio associato se questa best practice non fosse adottata: medio. La mancata implementazione di meccanismi di monitoraggio e notifica degli eventi appropriati può comportare l'impossibilità di rilevare i modelli di problemi, compresi quelli risolti mediante la correzione automatica. Un team verrà informato del degrado del sistema solo nel momento in cui gli utenti contattano il servizio clienti o per caso.

Guida all'implementazione

Quando si definisce una strategia di monitoraggio, un allarme attivato è un evento comune. Questo evento dovrebbe contenere un identificatore dell'allarme, lo stato dell'allarme (ad esempio `IN ALLARME` o `OK`) e dettagli su cosa l'ha innescato. In molti casi, è necessario rilevare un evento di allarme e inviare una notifica tramite e-mail. Questo è un esempio di azione su un allarme. La notifica degli allarmi è fondamentale per l'osservabilità, in quanto informa le persone giuste della presenza di un problema. Tuttavia, quando le operazioni eseguite sulla base degli eventi raggiungono un certo grado di maturità nella soluzione di osservabilità, è possibile risolvere automaticamente il problema senza la necessità dell'intervento umano.

Una volta stabiliti gli allarmi di monitoraggio dei KPI, è necessario inviare avvisi ai team appropriati quando vengono superate le soglie. Tali avvisi possono essere utilizzati anche per attivare processi automatizzati che tenteranno di porre rimedio al danno o alla compromissione.

Per un monitoraggio delle soglie più complesso, è necessario prendere in considerazione gli allarmi compositi. Gli allarmi compositi utilizzano una serie di allarmi di monitoraggio dei KPI per creare un avviso basato sulla logica di business operativa. Gli allarmi CloudWatch possono essere configurati

per l'invio di e-mail o per la registrazione di file di log nei sistemi di monitoraggio di terze parti tramite l'integrazione con Amazon SNS o Amazon EventBridge.

Passaggi dell'implementazione

Crea vari tipi di allarmi in base al modo in cui vengono monitorati i carichi di lavoro, ad esempio:

- Gli allarmi applicativi vengono utilizzati per rilevare quando una parte del carico di lavoro non funziona correttamente.
- Allarmi infrastrutturali indicano quando dimensionare le risorse. Gli allarmi possono essere visualizzati visivamente sui pannelli di controllo, essere inviati tramite Amazon SNS o tramite e-mail e utilizzati con Auto Scaling per aumentare o diminuire le risorse del carico di lavoro.
- Semplice allarmi statici per monitorare quando una metrica supera una soglia statica per un numero specificato di periodi di valutazione.
- Gli allarmi compositi possono tenere conto di allarmi complessi provenienti da più fonti.
- Una volta creato l'allarme è possibile generare eventi di notifica appropriati. Puoi richiamare direttamente una API Amazon SNS per inviare notifiche e collegare qualsiasi automazione per la correzione o la comunicazione.
- integra Amazon Health Aware per consentire il monitoraggio della visibilità sulle risorse AWS che potrebbero presentare un deterioramento. Per i carichi di lavoro aziendali essenziali, questa soluzione fornisce l'accesso ad avvisi proattivi e in tempo reale per i servizi AWS.

Risorse

Best practice Well-Architected correlate:

- Definizione di disponibilità

Documenti correlati:

- Creare un allarme CloudWatch basato su una soglia statica
- Che cos'è Amazon EventBridge?
- Che cos'è Amazon Simple Notification Service?
- Pubblicazione di parametri personalizzati
- Utilizzo degli allarmi di Amazon CloudWatch
- Amazon Health Aware (AHA)

- Configurazione degli allarmi compositi di CloudWatch
- Cosa c'è di nuovo in AWS Observability at re:Invent 2022

Strumenti correlati:

- CloudWatch
- CloudWatch X-Ray

REL11-BP07 Progettazione del prodotto in modo da soddisfare gli obiettivi di disponibilità e i contratti sul livello di servizio per i tempi di attività

Progetta il tuo prodotto in modo da soddisfare gli obiettivi di disponibilità e i contratti sul livello di servizio per i tempi di attività. Se pubblichi o accetti privatamente obiettivi di disponibilità o contratti sul livello di servizio per i tempi di attività, verifica che l'architettura e i processi operativi siano progettati in modo da supportarli.

Risultato desiderato: definizione per ogni applicazione di un obiettivo definito per la disponibilità e di un contratto sul livello di servizio per le metriche di prestazioni, che possono essere monitorati e gestiti per realizzare i risultati aziendali.

Anti-pattern comuni:

- Progettazione e implementazione di carichi di lavoro senza impostare alcun contratto sul livello di servizio.
- Impostazione di metriche elevate per il contratto sul livello di servizio senza fondamento logico o requisiti aziendali.
- Impostazione di contratti sul livello di servizio senza tenere conto delle dipendenze e dei relativi contratti sul livello di servizio sottostanti.
- Progettazione delle applicazioni senza tenere conto del Modello di responsabilità condivisa per la resilienza.

Vantaggi dell'adozione di questa best practice: la progettazione di applicazioni in base ai principali obiettivi di resilienza ti aiuta a realizzare gli obiettivi aziendali e a soddisfare le aspettative dei clienti. Questi obiettivi orientano un processo di progettazione delle applicazioni in grado di valutare diverse tecnologie e tenere conto di vari compromessi.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

La progettazione delle applicazioni deve tenere conto di una serie eterogenea di requisiti derivati da obiettivi aziendali, operativi e finanziari. Nell'ambito dei requisiti operativi, i carichi di lavoro devono avere obiettivi specifici in termini di metriche di resilienza, in modo da poter essere monitorati e supportati correttamente. Le metriche di resilienza non devono essere impostate o derivate dopo l'implementazione del carico di lavoro. Devono invece essere definite durante la fase di progettazione e contribuire a determinare i diversi compromessi e decisioni.

- Ogni carico di lavoro deve avere una serie di metriche di resilienza propria. Le metriche possono essere diverse da quelle di altre applicazioni aziendali.
- La riduzione delle dipendenze può avere un impatto positivo sulla disponibilità. Per ogni carico di lavoro è necessario considerare le dipendenze e i relativi contratti sul livello di servizio. In generale, seleziona dipendenze con obiettivi di disponibilità uguali o maggiori rispetto agli obiettivi del carico di lavoro.
- Prendi in considerazione progettazioni senza integrazioni serrate in modo che il carico di lavoro possa funzionare correttamente anche in caso di dipendenze compromesse, se possibile.
- Riduci le dipendenze del piano di controllo (control-plane), in particolare durante un ripristino o un peggioramento delle prestazioni. Valuta le progettazioni staticamente stabili per carichi di lavoro mission critical. Usa il contenimento delle risorse per aumentare la disponibilità delle dipendenze in un carico di lavoro.
- La visibilità e la strumentazione sono essenziali per soddisfare i contratti sul livello di servizio attraverso la riduzione del tempo medio di rilevamento (MTTD) e del tempo medio di ripristino (MTTR).
- Errori meno frequenti (tempo medio tra guasti, o MTBF, più lungo), tempi di rilevamento degli errori più brevi (MTTD minore) e tempi di riparazione più brevi (MTTR minore) sono i tre fattori usati per migliorare la disponibilità in sistemi distribuiti.
- La definizione e l'applicazione di metriche di resilienza per un carico di lavoro sono essenziali per qualsiasi progettazione efficace. Queste progettazioni devono tenere conto dei compromessi introdotti dalla complessità di progettazione, delle dipendenze dei servizi, delle prestazioni, del dimensionamento e dei costi.

Passaggi dell'implementazione

- Esamina e documenta la progettazione del carico di lavoro cercando di rispondere alle domande seguenti:

  - Dove vengono usati piani di controllo (control-plane) nel carico di lavoro?
  - Come viene implementata la tolleranza ai guasti nel carico di lavoro?
  - Quali sono i modelli di progettazione per dimensionamento, scalabilità automatica, ridondanza e componenti a disponibilità elevata?
  - Quali sono i requisiti per la coerenza e la disponibilità dei dati?
  - Vi sono aspetti da considerare in fatto di contenimento delle risorse o stabilità statica delle risorse?
  - Quali sono le dipendenze dei servizi?
- Definisci insieme agli stakeholder le metriche per il contratto sul livello di servizio in base all'architettura del carico di lavoro. Tieni conto dei contratti sul livello di servizio di tutte le dipendenze usate dal carico di lavoro.
- Una volta definiti gli obiettivi del contratto sul livello di servizio, ottimizza l'architettura in modo da soddisfare il contratto.
- Una volta impostata una progettazione che soddisfa il contratto sul livello di servizio, implementa modifiche operative, automazione dei processi e runbook anch'essi incentrati sulla riduzione dell'MTTD e dell'MTTR.
- Dopo aver implementato il contratto sul livello di servizio, devi monitorarlo e documentarlo.

Risorse

Best practice correlate:

- REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL12-BP05 Test della resilienza tramite l'utilizzo dell'ingegneria del caos
- REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati
- Comprendere lo stato del carico di lavoro

Documenti correlati:

- Disponibilità con ridondanza
- Principio dell'affidabilità: disponibilità

- Misurazione della disponibilità
- Limiti di isolamento dei guasti di AWS
- Modello di responsabilità condivisa per la resilienza
- stabilità statica utilizzando le zone di disponibilità
- Contratti sul livello di servizio AWS
- Linee guida per le architetture basate su celle su AWS
- Infrastruttura AWS
- Whitepaper sui modelli di resilienza multi-AZ avanzati

Servizi correlati:

- Amazon CloudWatch
- AWS Config
- AWS Trusted Advisor

REL 12. Come si testa l'affidabilità?

Dopo aver progettato il carico di lavoro in modo da essere resiliente alle sollecitazioni della produzione, i test sono l'unico modo per verificare il funzionamento corretto e offrire la resilienza prevista.

Best practice

- REL12-BP01 Utilizzo dei playbook per analizzare gli errori
- REL12-BP02 Esecuzione di analisi post-incidente
- REL12-BP03 Test dei requisiti funzionali
- REL12-BP04 Test dei requisiti di dimensionamento e prestazioni
- REL12-BP05 Test della resilienza tramite l'utilizzo dell'ingegneria del caos
- REL12-BP06 Esecuzione regolare di giornate di gioco

REL12-BP01 Utilizzo dei playbook per analizzare gli errori

Abilita risposte coerenti e tempestive a scenari di guasto che non sono ben compresi, documentando il processo di analisi nei playbook. I playbook sono le fasi predefinite eseguite per identificare i fattori che contribuiscono a uno scenario di guasto. I risultati provenienti da un passaggio del processo

vengono utilizzati per stabilire i passaggi successivi da intraprendere fino all'identificazione o alla risoluzione del problema.

Il playbook è una pianificazione proattiva che è necessario eseguire, in modo da potere intraprendere azioni reattive in modo efficace. Quando durante la produzione si verificano scenari di guasto non coperti dal playbook, risolvi innanzitutto il problema (spegni l'incendio). Quindi torna indietro e osserva le fasi intraprese per risolvere il problema e utilizzale per aggiungere una nuova voce al playbook.

Tieni presente che i playbook vengono utilizzati in risposta a specifici incidenti, mentre i runbook vengono utilizzati per ottenere esiti specifici. Spesso, i runbook vengono utilizzati per le attività di routine e i playbook vengono utilizzati per rispondere a eventi non di routine.

Anti-pattern comuni:

- Pianificare la distribuzione di un carico di lavoro senza conoscere i processi per diagnosticare i problemi o rispondere agli incidenti.
- Decisioni non pianificate sui sistemi da cui raccogliere log e parametri durante l'analisi di un evento.
- Non conservare parametri e eventi abbastanza a lungo da poter recuperare i dati.

Vantaggi dell'adozione di questa best practice: L'acquisizione di playbook garantisce l'esecuzione coerente dei processi. La codifica dei playbook limita l'introduzione di errori derivanti dall'attività manuale. L'automazione dei playbook riduce il tempo necessario per rispondere a un evento eliminando il requisito per l'intervento dei membri del team o fornendo loro informazioni aggiuntive quando inizia l'intervento.

Livello di rischio associato se questa best practice non fosse adottata: Alta

Guida all'implementazione

- Utilizza playbook per identificare i problemi. I playbook sono processi documentati per eseguire indagini sui problemi. Abilita risposte coerenti e tempestive agli scenari di errore documentando i processi nei playbook. I playbook devono contenere le informazioni e le istruzioni necessarie affinché una persona adeguatamente qualificata possa raccogliere le informazioni applicabili, identificare potenziali fonti di errore, isolare i guasti e stabilire i fattori che contribuiscono all'origine di un problema (eseguire l'analisi post-incidente).
  - Implementazione dei playbook come codice. Esegui le operazioni come codice mediante lo scripting dei playbook per assicurare coerenza e ridurre gli errori causati dai processi

manuali. I playbook possono essere composti da più script che rappresentano le diverse fasi che potrebbero essere necessarie per identificare i fattori che contribuiscono all'origine di un problema. Le attività dei runbook possono essere attivate o eseguite nell'ambito delle attività dei playbook oppure possono richiedere l'esecuzione di un playbook in risposta agli eventi identificati.

- Automazione dei playbook operativi con AWS Systems Manager
- AWS Systems Manager Run Command
- AWS Systems Manager Automation
- Cos'è AWS Lambda?
- Che cos'è Amazon EventBridge?
- Utilizzo degli allarmi di Amazon CloudWatch

Risorse

Documenti correlati:

- AWS Systems Manager Automation
- AWS Systems Manager Run Command
- Automazione dei playbook operativi con AWS Systems Manager
- Utilizzo degli allarmi di Amazon CloudWatch
- Utilizzo di Canary (Amazon CloudWatch Synthetics)
- Che cos'è Amazon EventBridge?
- Cos'è AWS Lambda?

Esempi correlati:

- Automating operations with Playbooks and Runbooks (Automazione delle operazioni con Playbook e Runbook)

REL12-BP02 Esecuzione di analisi post-incidente

Esamina gli eventi che influiscono sui clienti e identifica i fattori che vi hanno contribuito e gli elementi di azione preventivi. Utilizza queste informazioni per sviluppare modi per limitare o prevenire il ripetersi degli imprevisti. Sviluppa procedure per attivare risposte rapide ed efficaci.

Comunica i fattori che hanno contribuito al presentarsi dell'imprevisto e le azioni correttive secondo necessità, specificamente mirate per il pubblico di destinazione. All'occorrenza, adotta un metodo per comunicare queste cause ad altri.

Valuta perché i test esistenti non hanno individuato il problema. Aggiungi i test per questo caso se i test non esistono già.

Risultato desiderato: i tuoi team hanno un approccio coerente e concordato alla gestione dell'analisi post-incidente. Un meccanismo è il processo di correzione dell'errore (COE). Il processo COE aiuta i team a individuare, comprendere e gestire le cause principali degli incidenti, creando al contempo meccanismi e guardrail per limitare la probabilità che lo stesso incidente si ripeta.

Anti-pattern comuni:

- Individuare i fattori che hanno contribuito al verificarsi dell'incidente, ma non continuare a cercare in maniera più approfondita altri potenziali problemi e approcci da mitigare.
- Identificare le cause degli errori umani senza fornire alcuna formazione o automazione che potrebbe prevenirli.
- Concentrarsi sull'attribuzione delle colpe piuttosto che sulla comprensione della causa principale, creando così una cultura della paura e ostacolando la comunicazione costruttiva
- Mancata condivisione delle informazioni, che mantiene i risultati dell'analisi degli incidenti all'interno di un gruppo ristretto e impedisce ad altri di beneficiare delle lezioni apprese
- Nessun meccanismo che consenta di acquisire le conoscenze formali; in questo modo si perdono informazioni preziose in quanto non vengono preservate le lezioni apprese sotto forma di best practice aggiornate, con il conseguente rischio che gli incidenti si ripetano con la stessa causa principale o causa simile

Vantaggi dell'adozione di questa best practice: l'esecuzione di analisi post-incidente e la condivisione dei risultati consente ad altri carichi di lavoro di mitigare il rischio se hanno implementato gli stessi fattori che hanno contribuito al verificarsi dell'incidente e consente loro di implementare la mitigazione o il ripristino automatico prima che si verifichi un incidente.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

Una buona analisi post-incidente fornisce opportunità per proporre soluzioni comuni a problemi con modelli di architettura utilizzati in altri punti nei tuoi sistemi.

Un elemento fondamentale del processo COE è la documentazione e la risoluzione dei problemi. È consigliabile definire un modo standard per documentare le cause principali critiche e assicurarsi che queste vengano esaminate e risolte. Assegna in modo chiaro il responsabile del processo di analisi post-incidente. Designa un team o una persona responsabile della supervisione delle indagini e dei follow-up degli incidenti.

Promuovi una cultura basata sull'apprendimento e sul miglioramento piuttosto che sull'attribuzione di colpe. Insisti sul fatto che l'obiettivo è prevenire incidenti futuri e non penalizzare le persone.

Sviluppa procedure ben definite per l'esecuzione delle analisi post-incidente. Queste procedure dovrebbero stabilire le misure da adottare, le informazioni da raccogliere e le questioni chiave da risolvere durante l'analisi. Svolgi indagini approfondite sugli incidenti, andando oltre le cause immediate per identificare le cause principali e i fattori determinanti. Usa tecniche come i Cinque Perché per analizzare approfonditamente i problemi sottostanti.

Mantieni un archivio delle conclusioni derivanti dalle analisi degli incidenti. Queste conoscenze formali possono fungere da riferimento per futuri incidenti e attività di prevenzione. Condividi i risultati e gli approfondimenti delle analisi post-incidente e valuta la possibilità di organizzare riunioni di revisione post-incidente con invito aperto per discutere i risultati e le conclusioni.

Passaggi dell'implementazione

- Durante l'analisi post-incidente, assicurati che il processo non comporti la colpevolizzazione delle parti coinvolte. Ciò consente alle parti interessate di essere imparziali rispetto delle azioni correttive proposte, nonché di promuovere l'autovalutazione e la collaborazione a livello di team.
- Definisci una procedura standardizzata per documentare i problemi critici. Una struttura di esempio per tale documento è la seguente:
  - Cosa è successo?
  - Quale impatto ha avuto su clienti e attività?
  - Qual è stata la causa principale?
  - Di quali dati disponi a supporto di questo problema?
    - Ad esempio, metriche e grafici
  - Quali sono state le principali implicazioni sui pilastri critici, specialmente per quanto riguarda la sicurezza?
    - Quando progetti l'architettura dei carichi di lavoro, devi trovare dei compromessi tra i pilastri su cui si regge il contesto aziendale. Questo tipo di decisioni aziendali deve essere alla base delle tue priorità ingegneristiche. Potresti ridurre i costi a spese dell'affidabilità in ambienti

di sviluppo oppure, per quanto riguarda le soluzioni mission-critical, potresti ottimizzare l'affidabilità con costi maggiori. La sicurezza ha la massima priorità quando si tratta di proteggere i tuoi clienti.

  - Quali lezioni hai imparato?
  - Quali azioni correttive stai adottando?
    - Azioni correttive
    - Articoli correlati
- Crea precise procedure operative standard per lo svolgimento delle analisi post-incidente.
- Configura un processo standardizzato di segnalazione degli incidenti. Documenta in modo esaustivo tutti gli incidenti, includendo il rapporto iniziale sull'incidente, i log, le comunicazioni e le azioni intraprese durante l'incidente.
- Ricorda che un incidente non necessariamente comporta un'interruzione del servizio. Potrebbe trattarsi di un near miss o di un sistema che funziona in modo imprevisto pur continuando a svolgere la sua funzione aziendale.
- Migliora continuamente il processo di analisi post-incidente sulla base dei feedback e delle lezioni apprese.
- Acquisisci i risultati chiave in un sistema di gestione delle conoscenze e valuta eventuali modelli da aggiungere alle linee guide per gli sviluppatori o alle liste di controllo usate nella fase di pre-implementazione.

Risorse

Documenti correlati:

- Why you should develop a correction of error (COE) (Perché sviluppare una correzione dell'errore)

Video correlati:

- Amazon’s approach to failing successfully
- AWS re:Invent 2021 - Amazon Builders’ Library: Operational Excellence at Amazon

REL12-BP03 Test dei requisiti funzionali

Utilizza tecniche come i test unitari e i test di integrazione per convalidare le funzionalità richieste.

Puoi ottenere i migliori risultati quando questi test vengono eseguiti automaticamente come parte delle operazioni di sviluppo e distribuzione. Ad esempio, utilizzando AWS CodePipeline, gli sviluppatori affidano le modifiche a un repository di origine in cui CodePipeline rileva automaticamente le modifiche. Queste modifiche vengono create e vengono eseguiti test. Una volta completati i test, il codice creato viene distribuito ai server temporaneo per il test. Dal server temporaneo, CodePipeline esegue più test, come quelli di integrazione o caricamento. Una volta completati con successo i test, CodePipeline distribuisce il codice testato e approvato alle istanze di produzione.

Inoltre, l'esperienza dimostra che i test sintetici delle transazioni (noti anche come test canary, ma da non confondere con le implementazioni canary) in grado di eseguire e simulare il comportamento dei clienti sono uno dei processi di test più importanti. Esegui questi test costantemente sugli endpoint del carico di lavoro da diverse posizioni remote. Amazon CloudWatch Synthetics ti consente di creare "canary" per monitorare gli endpoint e le API.

Livello di rischio associato se questa best practice non fosse adottata: Alta

Guida all'implementazione

- Test dei requisiti funzionali. Includono test delle unità e test di integrazione che convalidano la funzionalità richiesta.
  - Utilizzo di CodePipeline con AWS CodeBuild per testare il codice ed eseguire compilazioni
  - AWS CodePipeline Adds Support for Unit and Custom Integration Testing with AWS CodeBuild (AWS CodePipeline aggiunge il supporto per i test di unità e integrazione personalizzati con AWS CodeBuild)
  - Distribuzione continua e integrazione continua
  - Utilizzo di Canary (Amazon CloudWatch Synthetics)
  - Automazione e test del software

Risorse

Documenti correlati:

- Partner APN: partner che possono essere d'aiuto nell'implementazione di una pipeline di integrazione continua

- AWS CodePipeline Adds Support for Unit and Custom Integration Testing with AWS CodeBuild (AWS CodePipeline aggiunge il supporto per i test di unità e integrazione personalizzati con AWS CodeBuild)
- Marketplace AWS: prodotti utilizzabili per l'integrazione continua
- Distribuzione continua e integrazione continua
- Automazione e test del software
- Utilizzo di CodePipeline con AWS CodeBuild per testare il codice ed eseguire compilazioni
- Utilizzo di Canary (Amazon CloudWatch Synthetics)

REL12-BP04 Test dei requisiti di dimensionamento e prestazioni

Utilizza tecniche come i test di carico per convalidare che il carico di lavoro soddisfi i requisiti di dimensionamento e prestazioni.

Nel cloud, puoi creare un ambiente di test su scala di produzione on demand per il tuo carico di lavoro. Se esegui questi test su un'infrastruttura ridotta, devi dimensionare i risultati osservati in base a ciò che pensi accadrà in produzione. I test di carico e prestazioni possono essere eseguiti anche in produzione se si fa attenzione a non influire sugli utenti effettivi e si contrassegna con tag i dati di test in modo da non utilizzare dati utente reali e non danneggiare le statistiche di utilizzo o i report di produzione.

Con i test, assicurati che le risorse di base, le impostazioni di dimensionamento, le quote di servizio e la progettazione di resilienza funzionino come previsto sotto carico.

Livello di rischio associato se questa best practice non fosse adottata: Alta

Guida all'implementazione

- Test dei requisiti di dimensionamento e prestazioni. Esegui test del carico per verificare che il carico di lavoro soddisfi i requisiti di dimensionamento e prestazioni.
  - Distributed Load Testing on AWS (Test di carico distribuito su AWS): simula migliaia di utenti connessi
  - Apache JMeter
    - Distribuisci la tua applicazione in un ambiente identico al tuo ambiente di produzione ed esegui un test di carico.
      - Utilizza un'infrastruttura come code concept per creare un ambiente il più simile possibile al tuo ambiente di produzione.

Risorse

Documenti correlati:

- Distributed Load Testing on AWS (Test di carico distribuito su AWS): simula migliaia di utenti connessi
- Apache JMeter

REL12-BP05 Test della resilienza tramite l'utilizzo dell'ingegneria del caos

Esegui regolarmente esperimenti di ingegneria del caos in ambienti di produzione o per quanto possibile ambienti analoghi per capire in che modo il sistema risponde a condizioni avverse.

Risultato desiderato:

La resilienza del carico di lavoro viene regolarmente verificata mediante l'applicazione dell'ingegneria del caos sotto forma di esperimenti di iniezione di errori o di inserimento di carichi imprevisti, nonché mediante il test della resilienza che convalida i comportamenti previsti noti del carico di lavoro durante un evento. Combina l'ingegneria del caos e i test della resilienza per verificare se il carico di lavoro è in grado di superare i guasti dei componenti ed eseguire il ripristino da interruzioni del servizio impreviste con un impatto minimo o nullo.

Anti-pattern comuni:

- Progettazione della resilienza, ma mancata verifica del funzionamento del carico di lavoro nel suo complesso in caso di errori.
- Mancata sperimentazione in scenari reali e con carichi previsti.
- Mancato trattamento degli esperimenti come codice o loro conservazione durante il ciclo di sviluppo.
- Mancata esecuzione degli esperimenti di ingegneria del caos sia nella pipeline CI/CD che esternamente alle implementazioni.
- Mancato utilizzo delle precedenti analisi post-incidente durante la determinazione degli errori su cui eseguire i test.

Vantaggi dell'adozione di questa best practice: l'introduzione di errori per verificare la resilienza del carico di lavoro consente di verificare che le procedure di ripristino della progettazione resiliente funzionerà se viene generato un vero e proprio errore.

Livello di rischio associato se questa best practice non fosse adottata: Medio

Guida all'implementazione

L'ingegneria del caos offre ai team la possibilità di continuare a inserire scenari di errore reali (simulazioni) in modo controllato a livello di fornitore di servizi, infrastruttura, carico di lavoro e componente con un impatto minimo o nullo per i clienti. Consente inoltre ai team di imparare dagli errori e osservare, misurare e migliorare la resilienza dei carichi di lavoro, nonché verificare l'attivazione degli avvisi e se tali avvisi vengono recapitati ai team se si verifica un evento definito.

Se applicata in modo continuativo, l'ingegneria del caos può mettere in evidenza i difetti del carico di lavoro che, se non risolti, possono avere ripercussioni negative sulla disponibilità e sulle operazioni.

> **Note**
>
> L'ingegneria del caos è la disciplina che sperimenta un sistema per creare fiducia nella capacità del sistema di affrontare condizioni turbolenti nella produzione. – Principi di ingegneria del caos

Se un sistema è in grado di sopportare queste interruzioni, l'esperimento di ingegneria del caos deve essere convertito in test automatico di regressione. In questo modo, gli esperimenti di ingegneria del caos devono essere eseguiti nell'ambito del ciclo di vita dello sviluppo dei sistemi (SDLC) e della pipeline CI/CD.

Per garantire che il carico di lavoro sia in grado di gestire un guasto del componente, esegui l'iniezione di eventi di errore reali durante l'esecuzione degli esperimenti. Ad esempio, esegui esperimenti relativi alla perdita di istanze Amazon EC2 o a eventi di failover delle istanze database Amazon RDS primario e quindi verifica che il carico di lavoro non sia stato compromesso oppure o che si stato interessato solo in minima parte. Utilizza una combinazione di errori dei componenti per simulare gli eventi che possono essere causati da un'interruzione del servizio in una zona di disponibilità.

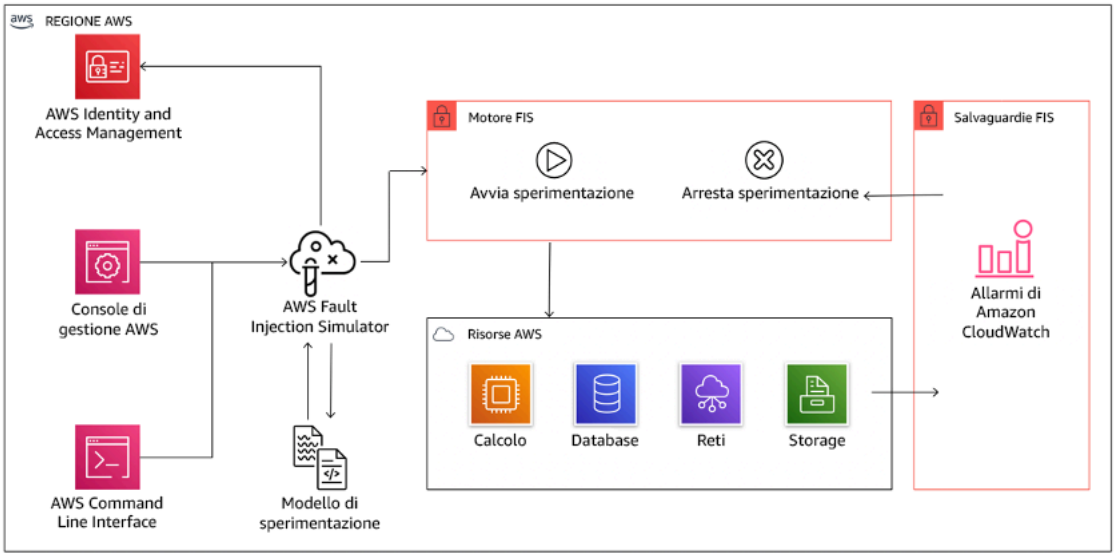
Per gli errori a livello di applicazione, ad esempio gli arresti anomali, puoi iniziare utilizzando fattori di stress, ad esempio l'esaurimento della memoria o della CPU.

Per convalidare i meccanismi di fallback o failover per le dipendenze esterne causate da interruzioni intermittenti dei servizi di rete, i componenti devono simulare tale evento bloccando l'accesso ai fornitori di terze parti per una durata specificata, che può durare da pochi secondi ad alcune ore.

Altre modalità di degrado possono causare funzionalità ridotte e risposte lente, spesso con conseguente interruzione dei servizi. Le fonti comuni di questo degrado sono una maggiore latenza nei servizi critici e una comunicazione di rete inaffidabile (pacchetti persi). Gli esperimenti basati su questi errori, inclusi gli effetti a livello di rete come latenza, messaggi eliminati ed errori DNS, possono prevedere l'incapacità di risolvere un nome, raggiungere il servizio DNS o stabilire connessioni a servizi dipendenti.

Strumenti dell'ingegneria del caos

AWS Fault Injection Service (AWS FIS) è un servizio completamente gestito per l'esecuzione di esperimenti di iniezione di errori che possono essere utilizzati come parte della pipeline di CD o al suo esterno. AWS FIS è una soluzione estremamente valida da utilizzare durante i giorni di gioco dell'ingegneria del caos. Supporta l'introduzione simultanea di errori in diversi tipi di risorse, ad esempio Amazon EC2, Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS) e Amazon RDS. Questi errori includono la cessazione delle risorse, la forzatura dei failover, l'applicazione di fattori di stress a CPU o memoria, la limitazione della lunghezza di banda della rete, la latenza e la perdita di pacchetti. Poiché è integrato con gli allarmi Amazon CloudWatch, è possibile impostare condizioni di arresto come guardrail per eseguire il rollback di un esperimento se causa un impatto inatteso.



AWS Fault Injection Service è integrato con le risorse AWS per consentire l'esecuzione di esperimenti di iniezione di errori per i carichi di lavoro.

Esistono anche diverse opzioni di terze parti per gli esperimenti di iniezione di errori. Queste includono strumenti open source, ad esempio Chaos Toolkit, Chaos Meshe Litmus Chaos, nonché opzioni commerciali come Gremlin. Per ampliare l'ambito degli errori che possono essere inseriti in

AWS, AWS FIS si integra con Chaos Mesh e Litmus Chaose ciò consente di coordinare i flussi di lavoro relativi all'iniezione di errori tra più strumenti. Ad esempio, puoi eseguire un test di stress sulla CPU di un pod utilizzando gli errori di Chaos Mesh o Litmus Chaos durante la cessazione di una percentuale casualmente selezionata di nodi di cluster mediante le operazioni di errore di AWS FIS.

Passaggi dell'implementazione

- Determinazione degli errori da utilizzare per gli esperimenti.

  Valutazione della progettazione del carico di lavoro a livello di resilienza. Tali progettazioni, create mediante le best practice del Canone di architettura AWS) giustificano i rischi in base alle dipendenze critiche, agli eventi pregressi, alle problematiche note e ai requisiti di conformità. Elenca i singoli elementi della progettazione che devono conservare la resilienza e gli errori per mitigare i quali è stata sviluppata. Per ulteriori informazioni su questi elenchi, consulta il whitepaper relativo alla prontezza operativa , contenente linee guida su come creare un processo per impedire che si verifichino di nuovo incidenti già noti. Il processo FMEA (Failure Modes and Effects Analysis) fornisce un framework per l'esecuzione di un'analisi degli errori a livello di componente e del relativo impatto sul carico di lavoro. Il processo FMEA è descritto più in dettaglio nell'articolo di Adrian Cockcroft su modalità di errore e resilienza continua.
- Assegna una priorità a ogni errore.

  Comincia con una categorizzazione approssimativa, ad esempio alta, media o bassa. Per assegnare la priorità, considera la frequenza dell'errore e l'impatto dell'errore sul carico di lavoro nel suo complesso.

  Durante la valutazione della frequenza di un errore specifico, analizza i precedenti dati per lo stesso carico di lavoro, se disponibili. Se non sono disponibili, utilizza i dati di altri carichi di lavoro eseguiti in un ambiente simile.

  Durante la valutazione dell'impatto di un errore specifico, in genere maggiore è l'ambito dell'errore, maggiore sarà l'impatto. Considera la progettazione e lo scopo del carico di lavoro. Ad esempio, la capacità di accedere ai datastore di origine è di cruciale importanza per un carico di lavoro responsabile della trasformazione e dell'analisi dei dati. In questo caso, darai la precedenza agli esperimenti relativi agli errori di accesso, nonché a quelli con accesso limitato a livello di larghezza di banda e inserimento di latenza.

  Le analisi post-incidente rappresentano un'ottima fonte di dati per la comprensione della frequenza e dell'impatto delle modalità di errore.

Utilizza la priorità assegnata per determinare il primo errore su cui eseguire l'esperimento e l'ordine in cui sviluppare i nuovi esperimenti di iniezione di errori.

- Per ogni esperimento eseguito, attieniti ai principi del volano dell'ingegneria del caos e della resilienza continua.

Volano dell'ingegneria del caos e della resilienza continua, che utilizza il metodo scientifico di Adrian Hornsby.

- Definisci lo stato stazionario come output misurabile di un carico di lavoro che indica un comportamento normale.

  Il carico di lavoro è associato allo stato stazionario se il suo funzionamento è affidabile e conforme a quanto previsto. Verifica pertanto che il carico di lavoro sia integro prima di definire lo stato stazionario. Lo stato stazionario non necessariamente indica l'assenza di impatto sul carico di lavoro se si verifica un errore in quanto una data percentuale di errori può rientrare nei limiti di valori accettabili. Lo stato stazionario rappresenta il punto di riferimento che verrà osservato

durante l'esperimento e che metterà in evidenza le anomalie se le ipotesi definite nel passaggio successivo non sono conformi alle previsioni.

Ad esempio, lo stato stazionario di un sistema di pagamento può essere definito come elaborazione di 300 TPS con una percentuale di successo pari al 99% e un tempo di round trip pari a 500 ms.

- Definisci un'ipotesi in merito alle reazioni del carico di lavoro all'errore.

Un'ipotesi ottimale fa riferimento al modo in cui il carico di lavoro presumibilmente è in grado di ridurre l'impatto dell'errore e salvaguardare lo stato stazionario. Nell'ipotesi è definito che, dato un errore di un tipo specifico, il sistema o il carico di lavoro rimarrà nello stato stazionario perché la progettazione del carico di lavoro ha previsto sistemi specifici di attenuazione degli errori. Il tipo di errore specifico e i sistemi di attenuazione devono essere specificati nell'ipotesi.

Per l'ipotesi è possibile utilizzare il seguente modello, anche se è accettabile una formulazione diversa:

> **Note**
>
> Se si verifica un *`errore specifico`* , il carico di lavoro *`nome del carico di lavoro`* descriverà *`i controlli di attenuazione`* per controbilanciare *`l'impatto sulle metriche aziendali o tecniche`*.

Ad esempio:

- In caso di arresto del 20% dei nodi nel gruppo di nodi Amazon EKS, l'API di creazione delle transazioni continua a servire il 99° percentile delle richieste in meno di 100 ms (stato stazionario). Verrà eseguito il ripristino dei nodi Amazon EKS entro cinque minuti; i pod verranno riprogrammati ed elaboreranno il traffico entro otto minuti dall'inizio dell'esperimento. Gli avvisi verranno attivati entro tre minuti.
- Se si verifica un errore in un'istanza Amazon EC2, il controllo dell'integrità Elastic Load Balancing del sistema degli ordini farà sì che Elastic Load Balancing si limiti a inviare richieste alle rimanenti istanze integre, mentre la funzionalità Amazon EC2 Auto Scaling sostituirà l'istanza in errore, garantendo un incremento inferiore allo 0,01% degli errori (5xx) lato server (stato stazionario).
- Se l'istanza database primario Amazon RDS restituisce un errore, il carico di lavoro della raccolta di dati della catena di approvvigionamento eseguirà il failover e si connetterà

all'istanza database in standby Amazon RDS per mantenere meno di un minuto di errori di lettura o scrittura del database (stato stazionario).

- Esegui l'esperimento inserendo l'errore.

  Per impostazione predefinita, un esperimento deve essere a prova di errore e tollerato dal carico di lavoro. Se sei consapevole del fatto che il carico di lavoro avrà esito negativo, non eseguire l'esperimento. L'ingegneria del caos deve essere utilizzata per individuare scenari noti sconosciuti o scenari completamente sconosciuti. "Scenari noti sconosciuti" fanno riferimento a quegli scenari di cui sei consapevole, ma non ne comprendi completamente la natura, mentre con "scenari completamente sconosciuti" si intendono quegli scenari a te non noti e di cui non ne comprendi la natura o i motivi. L'esecuzione di esperimenti su un carico di lavoro non funzionante non può fornire nuovi approfondimenti chiarificatori. L'esperimento deve infatti essere pianificato con attenzione, essere caratterizzato da un ambito ben definito relativamente al suo impatto, nonché fornire un meccanismo di rollback applicabile in caso di esiti negativi imprevisti. Se il criterio di due diligence indica che il carico di lavoro è in grado di sostenere l'esperimento, procedi ed esegui l'esperimento. Sono disponibili varie opzioni per l'inserimento degli errori. Per i carichi di lavoro in AWS, AWS FIS fornisce numerose simulazioni di errore predefinite denominate operazioni. Puoi anche definire operazioni personalizzate eseguibili in AWS FIS utilizzando i documenti AWS Systems Manager.

  È sconsigliato l'uso di script personalizzati per gli esperimenti di ingegneria del caos, a meno che gli script non siano in grado di rilevare lo stato corrente del carico di lavoro, generare log e fornire meccanismi di rollback e condizioni di arresto, laddove possibile.

  Un framework o set di strumenti efficace che supporta l'ingegneria del caos deve tenere traccia dello stato corrente di un esperimento, generare log e fornire meccanismi di rollback a supporto dell'esecuzione controllata di un esperimento. Inizia utilizzando un servizio noto, ad esempio AWS FIS, che consente di eseguire esperimenti con ambiti e meccanismi di sicurezza ben definiti in grado di eseguire il rollback dell'esperimento in caso di esiti negativi imprevisti. Per ulteriori informazioni sull'intera gamma di esperimenti che utilizzano AWS FIS, consulta anche la sezione relativa al laboratorio relativo alle app Well-Architected resilienti con ingegneria del caos. Inoltre, AWS Resilience Hub analizzerà il carico di lavoro e creerà gli esperimenti che potrai scegliere di implementare ed eseguire in AWS FIS.

> **Note**
>
> Per ogni esperimento, devi essere consapevole del suo ambito e del relativo impatto. È consigliabile eseguire la simulazione dell'errore in un ambiente non di produzione prima di eseguirla in un ambiente di produzione vero e proprio.

Gli esperimenti devono essere eseguiti in ambienti di produzione con un carico reale mediante implementazioni canary , che attivano sistemi sperimentali e di controllo, laddove possibile. L'esecuzione degli esperimenti durante gli orari non di punta è altamente consigliata al fine di ridurre al massimo potenziali eventi negativi durante la prima esecuzione dell'esperimento negli ambienti di produzione. Inoltre, se l'utilizzo dell'effettivo traffico clienti costituisce un rischio eccessivo, puoi eseguire gli esperimenti utilizzando una sintesi del traffico nell'infrastruttura di produzione utilizzando implementazioni sperimentali e di controllo. Se l'utilizzo di un ambiente di produzione non è possibile, esegui gli esperimenti in ambienti di pre-produzione il più simili possibile agli effettivi ambienti di produzione.

Devi definire e monitorare i guardrail per essere sicuro che l'esperimento non abbia un impatto sul traffico di produzione o sugli altri sistemi che superi i limiti accettabili. Definisci condizioni di arresto per interrompere l'esperimento se viene raggiunta la soglia definita nella metrica del guardrail. In tali condizioni devono essere incluse le metriche relative allo stato stazionario del carico di lavoro e le metriche riferite ai componenti in cui inserisci l'errore. Un monitor sintetico (definito anche canary utente) è una metrica che in genere deve essere inclusa come proxy utente. Le condizioni di arresto per AWS FIS sono supportate nel modello di esperimento, nella misura di un massimo di cinque condizioni di arresto per modello.

Uno dei principi dell'ingegneria del caos prevede la riduzione dell'ambito dell'esperimento e del relativo impatto.

Se da un lato deve essere prevista la possibilità di un determinato impatto negativo a breve termine, dall'altro il contenimento e la riduzione delle conseguenze negative degli esperimenti sono una responsabilità esclusiva dell'addetto all'ingegneria del caos.

Un metodo per verificare l'ambito e il potenziale impatto prevede l'esecuzione dell'esperimento dapprima in un ambiente non di produzione, la verifica che le soglie delle condizioni di arresto vengano attivate come previsto durante lo svolgimento di un esperimento e l'utilizzo effettivo

delle misure di osservabilità finalizzate all'acquisizione di un'eccezione, anziché eseguire l'esperimento direttamente in produzione.

Durante l'esecuzione di esperimenti di iniezione di errori, verifica che tutte le parti responsabili ne siano a conoscenza. Comunica ai team appropriati, ad esempio i team responsabili delle operazioni, dell'affidabilità dei servizi e del supporto clienti, quando verranno eseguiti gli esperimenti e l'impatto previsto. Metti a disposizione di questi team strumenti di comunicazione che consentano loro di informare i responsabili dell'esperimento di eventuali effetti avversi.

È necessario ripristinare lo stato originario del carico di lavoro e dei relativi sistemi sottostanti. La progettazione resiliente del carico di lavoro è spesso caratterizzata da funzionalità di riparazione automatica. Tuttavia, alcune progettazioni difettose o alcuni esperimenti non riusciti possono compromettere in modo imprevisto lo stato del carico di lavoro. Entro la fine dell'esperimento dovrai essere consapevole di questa situazione e ripristinare il carico di lavoro e i sistemi. Con AWS FIS puoi impostare una configurazione di rollback, definita anche post-operazione, all'interno dei parametri operativi. Una post-operazione ripristina una destinazione allo stato in cui si trovava prima dell'esecuzione dell'operazione stessa. Indipendentemente dal fatto che vengano eseguite in modalità automatica, ad esempio utilizzando AWS FIS, o manuale, queste post-operazioni devono essere incluse in un playbook in cui vengono descritte le procedure di rilevamento e gestione degli errori.

- Verifica l'ipotesi.

  Principi di ingegneria del caos è un documento contenente le linee guida su come verificare lo stato stazionario del carico di lavoro.

  È necessario concentrarsi sull'output misurabile di un sistema e non sugli attributi interni del sistema. Le misurazioni di tale output in un breve periodo di tempo costituiscono un'attestazione dello stato stazionario del sistema. La velocità di trasmissione effettiva del sistema nel suo complesso, le percentuali di errori e i percentili della latenza possono essere considerati metriche di interesse che rappresentano il comportamento di uno stato stazionario. Sulla base dei rilevamenti dei modelli di comportamento sistematico durante gli esperimenti, l'ingegneria del caos verifica che il sistema funzioni correttamente anziché tentare di convalidare il modo in cui funziona.

  Nei due esempi precedenti sono state incluse le metriche dello stato stazionario relative a un incremento inferiore allo 0,01% di errori (5xx) lato server e inferiore a un minuto di errori di lettura e scrittura del database.

Gli errori 5xx rappresentano una buona metrica perché sono la conseguenza della modalità di errore che un client del carico di lavoro sperimenterà direttamente. La misurazione degli errori del database risulta valida come conseguenza diretta dell'errore, ma deve essere supportata da una misurazione diretta dell'impatto, ad esempio le richieste cliente non riuscite o gli errori restituiti a livello di client. Includi anche un monitor sintetico, definito canary utente, in qualsiasi API o URI a cui il client del carico di lavoro ha accesso diretto.

- Migliora la progettazione del carico di lavoro con un occhio di riguardo per la resilienza.

  Se lo stato stazionario non è stato preservato, analizza in che modo puoi migliorare la progettazione del flusso di lavoro per azzerare l'impatto dell'errore applicando le best practice descritte nel Pilastro AWS Well-Architected relativo all'affidabilità. Ulteriori linee guida e risorse sono disponibili nella libreria di AWS Builder, dove sono contenuti articoli su come migliorare i controlli dell'integrità oppure impiegare nuovi tentativi con backoff nel codice dell'applicazione.

  Dopo aver implementato queste modifiche, esegui di nuovo l'esperimento (rappresentato dalla linea punteggiata nel volano relativo all'ingegneria del caos) per determinare la relativa efficacia. Se nella fase di verifica risulta che l'ipotesi è vera, il carico di lavoro sarà in stato stazionario e il ciclo continuerà.

- Esegui gli esperimenti con regolarità.

  Un esperimento di ingegneria del caos è un ciclo e gli esperimenti devono essere eseguiti regolarmente nell'ambito dell'ingegneria del caos. Se un carico di lavoro è conforme all'ipotesi dell'esperimento, l'esperimento deve essere automatizzato affinché venga eseguito continuamente come fase di regressione della pipeline CI/CD. Per ulteriori informazioni in merito, consulta questo blog relativamente alle procedure di esecuzione degli esperimenti AWS FIS utilizzando AWS CodePipeline. Questo laboratorio relativo a esperimenti AWS FIS ricorrenti in una pipeline CI/CD ti consente di fare esperienza pratica.

  Gli esperimenti di iniezione di errori fanno inoltre parte delle giornate di gioco (consulta REL12-BP06 Esecuzione regolare di giornate di gioco). Le giornate di gioco simulano un errore o un evento per verificare sistemi, processi e risposte dei team. Lo scopo è di eseguire effettivamente le azioni che compirebbe il team come se si verificasse un evento eccezionale.

- Acquisisci e archivia i risultati degli esperimenti.

  I risultati degli esperimenti di iniezione di errori devono essere acquisiti e resi persistenti. Includi tutti i dati necessari, ad esempio orari, carico di lavoro e condizioni, in modo da essere in grado di analizzare i risultati e i trend in un secondo momento. I risultati potrebbero includere, ad esempio,

screenshot dei pannelli di controllo, dump in formato CSV del database delle metriche oppure appunti scritti a mano relativi a eventi e osservazioni associati all'esperimento. La registrazione degli esperimenti mediante AWS FIS può rientrare nel processo di acquisizione dei dati.

Risorse

Best practice correlate:

- REL08-BP03 Esecuzione di test di resilienza come parte integrante dell'implementazione
- REL13-BP03 Esecuzione di test sull'implementazione del ripristino di emergenza per convalidare l'implementazione

Documenti correlati:

- What is AWS Fault Injection Service? (Che cos'è AWS Fault Injection Service?)
- What is AWS Resilience Hub? (Che cos'è AWS Resilience Hub?)
- Principi di ingegneria del caos
- Chaos Engineering: Planning your first experiment (Ingegneria del caos: pianificazione del primo esperimento)
- Resilience Engineering: Learning to Embrace Failure
- Chaos Engineering stories (Storie relative all'ingegneria del caso)
- Evitare fallback nei sistemi distribuiti
- Canary Deployment for Chaos Experiments (Implementazione canary per gli esperimenti di ingegneria del caos)

Video correlati:

- AWS re:Invent 2020: Testing resiliency using chaos engineering (ARC316) (Esecuzione di test di resilienza mediante l'ingegneria del caos [ARC316])
- AWS re:Invent 2019: migliorare la resilienza con l'ingegneria del caos (DOP309-R1)
- AWS re:Invent 2019: Performing chaos engineering in a serverless world (CMY301) (Esecuzione dell'ingegneria del caos in uno scenario serverless [CMY301])

Esempi correlati:

- Well-Architected lab: Level 300: Testing for Resiliency of Amazon EC2, Amazon RDS, and Amazon S3 (Test della resilienza di Amazon EC2, Amazon RDS e Amazon S3)
- Chaos Engineering on AWS lab (Laboratorio relativo all'ingegneria del caos in AWS)
- Resilient and Well-Architected Apps with Chaos Engineering lab (Laboratorio relativo alle app Well-Architected resilienti con ingegneria del caos)
- Serverless Chaos lab (Laboratorio relativi a esperimenti di ingegneria del caos per architetture serverless)
- Measure and Improve Your Application Resilience with AWS Resilience Hub lab (Laboratorio di misurazione e ottimizzazione della resilienza dell'applicazione con AWS Resilience Hub)

Strumenti correlati:

- AWS Fault Injection Service
- Marketplace AWS: Gremlin Chaos Engineering Platform (Piattaforma di ingegneria del caos di Gremlin)
- Chaos Toolkit
- Chaos Mesh
- Litmus

REL12-BP06 Esecuzione regolare di giornate di gioco

Utilizza le giornate di gioco per provare regolarmente le procedure per rispondere a eventi ed errori nel modo più vicino possibile alla produzione (anche negli ambienti di produzione) con le persone che si occuperanno di eventuali scenari di errore reali. Le giornate di gioco applicano misure per garantire che gli eventi di produzione non influiscano sugli utenti.

Le giornate di gioco simulano un errore o un evento per testare sistemi, processi e risposte dei team. Lo scopo è di eseguire effettivamente le azioni che compirebbe il team come se si verificasse un evento eccezionale. Questi ti aiuta a capire dove puoi apportare dei miglioramenti e ti può aiutare a sviluppare un'esperienza organizzativa nella gestione degli eventi. Tali azioni devono essere svolte regolarmente in modo che il team costruisca una memoria muscolare su come rispondere.

Quando la progettazione per la resilienza è in loco ed è stata testata in ambienti non di produzione, un game day è il modo per garantire che tutto funzioni come pianificato in produzione. Una giornata di gioco, soprattutto la prima, è un'attività di duro lavoro per tutti, in cui tutti gli ingegneri e i team operativi vengono informati in merito a quando accadrà e cosa accadrà. I runbook sono in loco.

Gli eventi simulati, compresi i possibili eventi di guasto, vengono eseguiti nei sistemi di produzione nel modo prescritto e ne viene valutato l'impatto. Se tutti i sistemi funzionano come progettato, il rilevamento e la correzione automatica avvengono con un impatto minimo o nullo. Tuttavia, se si osserva un impatto negativo, viene eseguito il rollback del test e i problemi relativi al carico di lavoro vengono risolti, se necessario manualmente (utilizzando il runbook). Poiché le giornate di gioco hanno spesso luogo in produzione, è necessario prendere tutte le precauzioni per garantire che non vi sia alcun impatto sulla disponibilità per i clienti.

Anti-pattern comuni:

- Documentare le procedure senza mai esercitarle.
- Non includere i responsabili delle decisioni aziendali negli esercizi di test.

Vantaggi dell'adozione di questa best practice: Eseguire giornate di gioco garantisce che tutto il personale segua le policy e le procedure quando si verifica un incidente reale e convalida che tali policy e procedure siano appropriate.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

- Programma giornate di gioco per provare regolarmente i tuoi runbook e playbook. Le giornate di gioco devono coinvolgere tutte le persone implicate in un evento di produzione: proprietari di aziende, personale addetto allo sviluppo, personale operativo e team di risposta agli incidenti.
  - Esegui i test di carico o delle prestazioni e successivamente esegui l'iniezione degli errori.
  - Ricerca anomalie nei tuoi runbook e opportunità di provare i tuoi playbook.
    - In caso di deviazione dai tuoi runbook, perfeziona il runbook o correggi il comportamento. Se ti eserciti sul tuo playbook, identifica il runbook che avrebbe dovuto essere usato, oppure creane uno nuovo.

Risorse

Documenti correlati:

- Che cos'è AWS GameDay?

Video correlati:

- AWS re:Invent 2019: migliorare la resilienza con l'ingegneria del caos (DOP309-R1)

Esempi correlati:

- AWS Well-Architected Labs – Test di resilienza

REL 13. Come si pianifica il disaster recovery o ripristino di emergenza?

Avere backup e componenti del carico di lavoro ridondanti in loco è l'inizio della strategia di disaster recovery. RTO e RPO sono i tuoi obiettivi per il ripristino del carico di lavoro. Imposta questi valori in base alle esigenze aziendali. Implementa una strategia per raggiungere questi obiettivi, prendendo in considerazione le posizioni e la funzione delle risorse e dei dati del carico di lavoro. La probabilità di interruzione e il costo del ripristino sono fattori chiave che aiutano a comunicare il valore aziendale che può avere il ripristino di emergenza per un carico di lavoro.

Best practice

- REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati
- REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino
- REL13-BP03 Esecuzione di test sull'implementazione del ripristino di emergenza per convalidare l'implementazione
- REL13-BP04 Gestione della deviazione di configurazione nel sito o nella Regione del ripristino di emergenza
- REL13-BP05 Automatizzazione del ripristino

REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati

Il carico di lavoro ha un Recovery Time Objective (RTO) e Recovery Point Objective (RPO).

Il Recovery Time Objective (RTO) è il ritardo massimo accettabile tra l'interruzione del servizio e il suo ripristino. Questo determina ciò che viene considerato un intervallo di tempo accettabile quando il servizio non è disponibile.

Recovery Point Objective (RPO) è il periodo di tempo massimo accettabile dall'ultimo punto di ripristino dei dati. Questo determina ciò che viene considerato una perdita di dati accettabile tra l'ultimo punto di ripristino e l'interruzione del servizio.

RTO e RPO sono valori importanti quando si seleziona una strategia adeguata di ripristino di emergenza per il proprio carico di lavoro. Tali obiettivi sono stabiliti dall'azienda e poi vengono utilizzati dai team tecnici per selezionare e implementare una strategia di ripristino di emergenza.

Risultato desiderato:

Ogni carico di lavoro ha un RTO e un RPO assegnati, definiti in base all'impatto aziendale. Il carico di lavoro viene assegnato a un livello predefinito, che stabilisce la disponibilità del servizio e la perdita accettabile di dati, con un RTO e un RPO associati. Se tale livello non è raggiungibile, è possibile assegnare un livello personalizzato per carico di lavoro, con l'obiettivo di creare i livelli in un secondo momento. RTO e RPO sono valori fondamentali per la selezione di una strategia di ripristino di emergenza da implementare per il carico di lavoro. Altre riflessioni nel momento della scelta di una strategia di ripristino di emergenza sono i vincoli economici, le dipendenze del carico di lavoro e i requisiti operativi.

Per l'RTO è necessario comprendere l'impatto in base alla durata di un'interruzione. È lineare o ci sono implicazioni non lineari? (Ad esempio, dopo 4 ore, chiudi una linea di produzione fino l'inizio del turno successivo).

Una matrice di ripristino di emergenza, come quella seguente, può aiutarti a capire come la criticità del carico di lavoro sia collegata agli obiettivi di ripristino. (Da notare che i valori reali per gli assi X e Y devono essere personalizzati in base alle esigenze della tua organizzazione).

Matrice di ripristino di emergenza						
		Obiettivo del punto di ripristino				
		meno di 1 minuto	meno di 1 ora	meno di 6 ore	meno di 1 giorno	Più di 1 giorno
Obiettivo del tempo di ripristino	meno di 10 minuti	Critica	Critica	Alta	Medio	Medio
	meno di 2 ore	Critica	Alta	Medio	Medio	Bassa
	meno di 8 ore	Alta	Medio	Medio	Bassa	Bassa
	meno di 24 ore	Medio	Medio	Bassa	Bassa	Bassa
	Più di 24 ore	Medio	Bassa	Bassa	Bassa	Bassa

Figura 16: Matrice di ripristino di emergenza

Anti-pattern comuni:

- Nessun obiettivo di ripristino definito.

- Selezione di obiettivi di ripristino arbitrari.
- Selezione di obiettivi di ripristino troppo tolleranti e che non soddisfano gli obiettivi di business.
- Mancanza di comprensione dell'impatto dei tempi di inattività e perdita dei dati.
- Selezione di obiettivi di ripristino non realistici, come tempo zero di ripristino e nessuna perdita di dati, che potrebbero non essere raggiungibili per la configurazione del tuo carico di lavoro.
- Selezione di obiettivi di ripristino più severi rispetto agli obiettivi aziendali effettivi. Questo costringe a effettuare implementazioni di ripristino di emergenza più costose e complicate rispetto alle esigenze del carico di lavoro.
- Selezione di obiettivi di ripristino non compatibili con quelli di un carico di lavoro dipendente.
- I tuoi obiettivi di ripristino non considerano i requisiti di conformità normativa.
- RTO e RPO definiti per un carico di lavoro, ma mai testati.

Vantaggi dell'adozione di questa best practice: Gli obiettivi di ripristino in termini di tempo e perdita di dati sono necessari per guidare l'implementazione del disaster recovery.

Livello di rischio associato se questa best practice non fosse adottata: Alta

Guida all'implementazione

Per un dato carico di lavoro devi considerare l'impatto dei tempi di inattività e della perdita dei dati per la tua azienda. L'impatto generalmente aumenta all'aumentare dei tempi di inattività o della perdita dei dati, ma il ritmo di tale crescita cambia in base al tipo di carico di lavoro. Ad esempio, potresti tollerare l'inattività per massimo un'ora con conseguenze minime, ma successivamente l'impatto diventerebbe rapidamente più serio. L'impatto sull'azienda si manifesta in forme diverse, tra cui costi economici (come perdita di fatturato), fiducia del cliente (e impatto sulla reputazione), problematiche operative (come stipendi in ritardo o diminuzione della produttività) e rischi normativi. Usa i passaggi seguenti per comprendere questi aspetti e impostare i valori RTO e RPO per il tuo carico di lavoro.

Passaggi dell'implementazione

1. Individua gli stakeholder aziendali per questo carico di lavoro e collabora con loro per implementare questi passaggi. Gli obiettivi di ripristino di un carico di lavoro sono il frutto di una decisione aziendale. I team tecnici, quindi, lavorano con gli stakeholder aziendali e usano questi obiettivi per selezionare una strategia di ripristino di emergenza.

> **Note**
>
> Per i passaggi 2 e 3 puoi usare the section called “Foglio di lavoro di implementazione”.

2. Raccogli le informazioni necessarie per prendere una decisione rispondendo alle domande qui di seguito.
3. Hai categorie o livelli di criticità in termini di impatto del tuo carico di lavoro nella tua organizzazione?
   a. Se sì, assegna questo carico di lavoro a una categoria
   b. Se no, definisci queste categorie. Crea al massimo cinque categorie e perfeziona l'intervallo del tuo Obiettivo del tempo di ripristino (RTO) per ognuna. Ecco alcune categorie di esempio: critico, alto, medio, basso. Per capire come mappare i carichi di lavoro rispetto alle categorie devi considerare se il carico di lavoro è mission-critical, importante per l'azienda o non trainante.
   c. Imposta i valori RTO e RPO del carico di lavoro in base alla categoria. Scegli sempre una categoria più severa (RTO e RPO inferiori) rispetto ai valori grezzi calcolati in questa fase. Se ciò comporta una variazione significativa di valore non rispondente alle esigenze, prendi in considerazione la possibilità di creare una nuova categoria.
4. In base alle risposte assegna i valori RTO e RPO al carico di lavoro. Puoi farlo direttamente o assegnando il carico di lavoro a un livello predefinito di servizio.
5. Crea un documento con il piano di ripristino di emergenza (DRP) per questo carico di lavoro, che sarà parte del piano di continuità aziendale della tua organizzazione (BCP), in un punto accessibile al team del carico di lavoro e agli stakeholder.
   a. Registra i valori RTO e RPO e le informazioni usate per definire questi valori. Includi la strategia utilizzata per valutare l'impatto del carico di lavoro sull'azienda.
   b. Registra altre metriche, oltre ai valori RTO e RPO che stai monitorando o che pensi di monitorare per gli obiettivi di ripristino di emergenza.
   c. Dopo aver creato questi valori, potrai aggiungere i dettagli della tua strategia di ripristino di emergenza e il runbook.
6. Osservando le criticità del carico di lavoro in una matrice come quella della Figura 15, puoi iniziare a stabilire livelli predefiniti di servizio per la tua organizzazione.
7. Dopo aver implementato una strategia di ripristino di emergenza (o un proof of concept per una strategia di ripristino di emergenza) secondo quanto stabilito da the section called “REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino”, testa questa strategia per stabilire i valori reali di RTC (Recovery Time Capability) e di RPC (Recovery Point

Capability) del carico di lavoro. Se questi valori non sono in linea con gli obiettivi target di ripristino, puoi collaborare con gli stakeholder della tua azienda per modificarli o cambiare la strategia di ripristino di emergenza in modo che possa soddisfare tali obiettivi.

Domande principali

1. Qual è il tempo massimo durante il quale il carico di lavoro può essere inattivo prima che questo abbia un impatto grave sull'attività?
   a. Definisci il costo monetario (impatto finanziario diretto) sull'attività al minuto se il carico di lavoro è inattivo.
   b. Considera che l'impatto non è sempre lineare. L'impatto può essere limitato all'inizio e poi aumentare rapidamente oltre un punto critico specifico.
2. Qual è la quantità massima di dati che possiamo perdere prima che questo abbia un impatto grave sull'attività?
   a. Considera questo valore per gli archivi di dati più strategici. identifica le criticità relative ad altri archivi di dati.
   b. I dati del carico di lavoro possono essere ricreati se persi? Se questo è operativamente più facile rispetto al backup e al ripristino, scegli il valore RPO in base alla criticità dei dati di origine utilizzati per ricreare i dati del carico di lavoro.
3. Quali sono gli obiettivi di ripristino e le aspettative di disponibilità dei carichi di lavoro da cui questo valore dipende (downstream) o i carichi di lavoro che dipendono da questo valore (upstream)?
   a. Scegli obiettivi di ripristino che consentono a questo carico di lavoro di soddisfare i requisiti delle dipendenze upstream.
   b. Scegli obiettivi di ripristino che sono raggiungibili considerate le funzionalità di ripristino delle dipendenze downstream. Possono essere escluse le dipendenze downstream non critiche (quelle che puoi "aggirare"). In alternativa, lavora con dipendenze downstream critiche per migliorare le funzionalità di ripristino, laddove necessario.

Domande aggiuntive

Considera queste domande e come possono essere applicate a questo carico di lavoro:

4. Hai RTO e RPO diversi a seconda del tipo di interruzione (Regione rispetto ad AZ e così via)?
5. Esiste un periodo specifico (stagionalità, eventi commerciali, lanci di prodotto) in cui RTO/RPO possono cambiare? Se sì, qual è la misurazione diversa e il vincolo temporale?

6. Se il carico di lavoro viene perturbato, quanti clienti ne subiranno l'impatto?
7. Qual è l'impatto sulla reputazione se il carico di lavoro è perturbato?
8. Quali altri impatti operativi possono verificarsi se il carico di lavoro subisce perturbazioni? Ad esempio, l'impatto sulla produttività dei dipendenti se i sistemi e-mail non sono disponibili o sei i sistemi di buste paga non sono in grado di inviare le transazioni.
9. In che modo il carico di lavoro e i valori RTO e RPO si allineano alla linea di business e alla strategia di ripristino di emergenza dell'organizzazione?
10. Esistono obblighi contrattuali interni per fornire un servizio? Esistono delle penali nel caso in cui non siano soddisfatti?
11. Quali sono i limiti normativi o di conformità dei dati?

Foglio di lavoro di implementazione

Puoi usare questo foglio di lavoro per le fasi 2 e 3 dell'implementazione. Adegua questo foglio di lavoro in base alle tue esigenze specifiche, aggiungendo, ad esempio, altre domande.

Passo 2: domande principali	Si applica al carico di lavoro?	RTO del carico di lavoro	RPO del carico di lavoro	RTO rettif.	RPO rettif.	Istruzioni
[1] tempo massimo di inattività del carico di lavoro						misurato in tempo dall'inizio del malfunzionamento al ripristino
[2] quantità massima di dati che possono essere persi						misurato in tempo trascorso dall'ultimo set di dati integro ripristinabile
[3a] dipendenze a monte						inserire gli obiettivi di recupero a monte più rigorosi
[3b] riconciliazione delle dipendenze a valle						inserire gli obiettivi di recupero a valle meno rigorosi
[3a] riconciliazione delle dipendenze a monte						Se il valore a monte è inferiore ai valori attuali e il valore a valle è superiore,
[3b] riconciliazione delle dipendenze a valle						operare sulle dipendenze per riconciliare i valori e inserirli qui.
[3] dipendenze						ridurre i valori per soddisfare le dipendenze a monte o alzarli in base alle capacità delle dipendenza a valle
Passo 2: domande aggiuntive						Indicare se la domanda è pertinente. Saltarla in caso affermativo
RTO/RPO di base						Riportare qui i valori di RTO e RPO sopra indicati
[4] tipo di malfunzionamento	[ ]S / [ ]N					Inserire gli obiettivi di recupero per i tipi di evento con i requisiti più rigorosi
[5] obiettivi specifici basati sul tempo	[ ]S / [ ]N					Inserire gli obiettivi di recupero per i tempi con i requisiti più rigorosi
[6] clienti che sperimentano il disservizio	[ ]S / [ ]N					Tracciare un grafico dei clienti che sperimentano il disservizio in funzione del tempo di inattività o dei dati persi. Utilizzare tale grafico per inserire i valori massimi di RTO e RPO ammissibili in base all'impatto sui clienti
[7] impatto reputazionale	[ ]S / [ ]N					Lavorare in modo congiunto con l'azienda per determinare i massimi valori di RTO e RPO in base all'impatto sulla reputazione
[8] impatto operativo	[ ]S / [ ]N					Inserire i valori massimi di RTO e RPO sulla base dell'impatto operativo
[9] allineamento aziendale	[ ]S / [ ]N					Inserire i valori massimi di RTO e RPO per i carichi di lavoro di questo tipo in base ai requisiti LOB e organizzativi
[10] obblighi contrattuali	[ ]S / [ ]N					Inserire i valori massimi di RTO e RPO sulla base degli obblighi contrattuali
[11] conformità normativa	[ ]S / [ ]N					Inserire i valori massimi di RTO e RPO sulla base delle norme di conformità applicabili
obiettivo sulla base delle domande aggiuntive						Selezionare il valore minimo (valore più rigoroso) dalle domande 4-11 e inserirlo qui
obiettivo rettificato						Se non è possibile raggiungere gli obiettivi indicati nella riga precedente, collaborare con le parti interessate per allentare i vincoli e inserire un nuovo minimo qui.
RTO/RPO rettificato						Inserire il valore inferiore tra RPO/RTO di base e valore obiettivo rettificato
Passo 3						
Mappatura su categorie o livelli predefiniti						Regolare entrambi i valori verso il basso (requisito più rigoroso) per allinearsi al livello più vicino definito

Foglio di lavoro

Livello di impegno per il piano di implementazione: Bassa

Risorse

Best practice correlate:

- the section called “REL09-BP04 Ripristino periodico dei dati per verificare l'integrità e i processi di backup:”
- the section called “REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino”
- the section called “REL13-BP03 Esecuzione di test sull'implementazione del ripristino di emergenza per convalidare l'implementazione”

Documenti correlati:

- AWS Architecture Blog: Disaster Recovery Series
- Ripristino di emergenza dei carichi di lavoro su AWS: ripristino nel cloud (whitepaper di AWS)
- Gestire le policy di resilienza con AWS Resilience Hub
- Partner APN: partner che possono assistere con disaster recovery
- Marketplace AWS: prodotti utilizzabili per il disaster recovery

Video correlati:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (Modelli architetturali per applicazioni attive-attive su più Regioni) (ARC209-R2)
- Ripristino di emergenza di carichi di lavoro su AWS

REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino

Definisci una strategia di ripristino di emergenza (DR) che soddisfi gli obiettivi di ripristino del carico di lavoro. Scegli una strategia, ad esempio backup e ripristino, standby (attivo/passivo) o attivo/attivo.

Risultato desiderato: definizione e implementazione di una strategia di ripristino di emergenza per ogni carico di lavoro che permette al carico di lavoro di realizzare gli obiettivi di ripristino di emergenza. Le strategie di ripristino di emergenza tra carichi di lavoro utilizzano modelli riutilizzabili (come strategie descritte in precedenza),

Anti-pattern comuni:

- Implementazione di procedure di ripristino incoerenti per carichi di lavoro con obiettivi di ripristino simili.
- Implementazione di una strategia di ripristino di emergenza ad-hoc quando si verifica un disastro.
- Assenza di piani per il ripristino di emergenza.
- Dipendenza dalle operazioni del piano di controllo durante il ripristino.

Vantaggi dell'adozione di questa best practice:

- L'utilizzo di strategie di ripristino definite consente di utilizzare strumenti e procedure di test comuni.
- L'uso di strategie di ripristino definite permette la condivisione delle informazioni tra team e l'implementazione del ripristino di emergenza nei carichi dl lavoro di loro proprietà.

Livello di rischio associato alla mancata adozione di questa best practice: elevato Senza una strategia di ripristino di emergenza pianificata, implementata e testata, è poco probabile riuscire a raggiungere gli obiettivi di ripristino in caso di eventi disastrosi.

Guida all'implementazione

Una strategia di ripristino di emergenza si basa sulla capacità di creare il tuo carico di lavoro in un sito di ripristino se la tua sede principale non è disponibile per eseguire il carico di lavoro. Gli obiettivi di ripristino più comuni sono RTO e RPO, come discusso in REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati.

Una strategia di ripristino di emergenza (DR) su più zone di disponibilità (AZ) all'interno di un singolo Regione AWS può offrire la mitigazione rispetto a eventi disastrosi come incendi, alluvioni e interruzioni gravi dell'energia. Se è un requisito implementare una protezione rispetto a un evento improbabile che impedisca al tuo carico di lavoro di poter essere eseguito in un determinato Regione AWS, puoi usare una strategia di ripristino di emergenza basata su più regioni.

Quando pianifichi una strategia di ripristino di emergenza su più regioni, devi scegliere una delle seguenti strategie. Sono elencate in ordine crescente di complessità e di costi e in ordine decrescente di RTO e RPO. Per regione di ripristino si intende una Regione AWS diversa da quella primaria usata per il carico di lavoro.

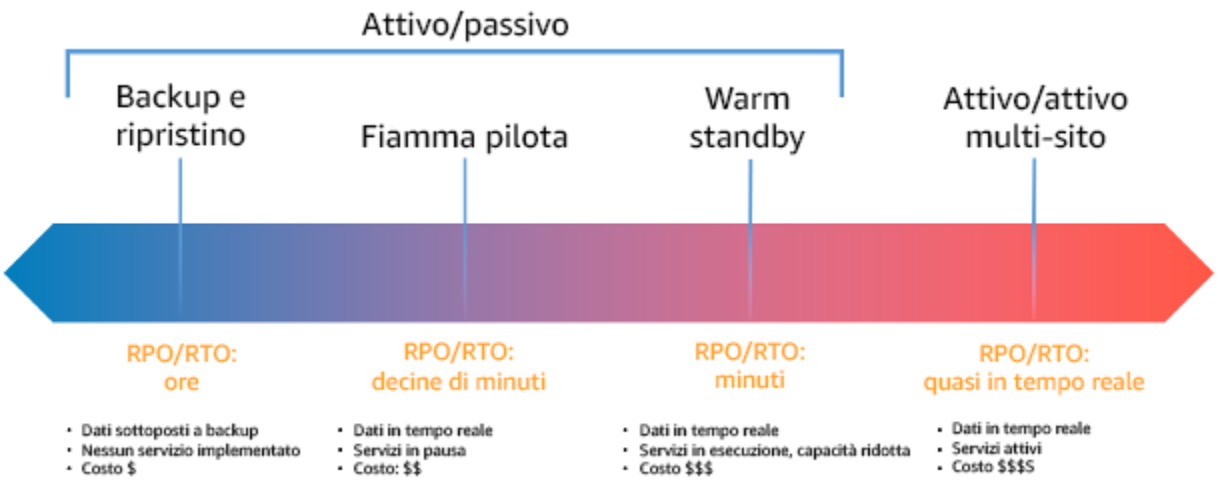


Figura 17: Strategie di ripristino di emergenza (DR)

- Backup e ripristino (RPO nell'ordine di ore, RTO in 24 ore o meno): backup dei dati e delle applicazioni nella regione di ripristino. Adottando backup continui o automatizzati otterrai un ripristino point-in-ime che può ridurre il valore dell'RPO fino a raggiungere in alcuni casi 5 minuti. Nel caso in cui si verifichi un disastro, distribuirai l'infrastruttura (usando l'infrastruttura come codice per ridurre l'RTO), distribuirai il codice e ripristinerai i dati del backup dopo un disastro nella regione di ripristino.
- Pilot Light (RPO nell'ordine di minuti, RTO nell'ordine di decine di minuti): provisioning di una copia dell'infrastruttura principale del carico di lavoro nella regione di ripristino. Replica i dati nella regione di ripristino e crea un backup in essa. Le risorse necessarie per supportare la replica dei dati e il backup, come database e archiviazione di oggetti, sono sempre attive. Altri elementi come i server applicativi o il calcolo serverless non vengono distribuiti, ma possono essere creati quando necessari con la configurazione e il codice applicativo richiesti.
- Warm Standby (RPO nell'ordine di secondi, RTO nell'ordine di minuti): esecuzione continua di una versione ridotta ma completamente funzionale del carico di lavoro nella regione di ripristino. I sistemi business critical sono completamente duplicati e sono sempre accesi, ma con un parco istanze ridimensionato. I dati vengono replicati e si trovano nella regione di ripristino. Quando viene il momento del ripristino, il sistema viene dimensionato rapidamente per gestire il carico di produzione. Maggiore è il dimensionamento nella strategia di Warm Standby, più bassi saranno l'RTO e la dipendenza del piano di controllo (control-plane). Quando il dimensionamento è completo, si parla di standby a caldo.

- Attivo/attivo multi-regione (multisito) (RPO quasi pari a zero, RTO potenzialmente pari a zero): il carico di lavoro viene implementato in più regioni AWS e distribuisce attivamente il traffico da più Regioni AWS. Questa strategia comporta la sincronizzazione dei dati tra le regioni. È necessario evitare o gestire possibili conflitti causati da scritture sullo stesso record in due diverse repliche regionali, un'attività che potrebbe rivelarsi complessa. La replica dei dati è utile per la sincronizzazione dei dati e ti proteggerà da alcuni tipi di disastri, ma non dalla corruzione o dalla distruzione dei dati, a meno che la tua soluzione non includa opzioni per il ripristino point-in-time.

> **Note**
>
> La differenza tra Pilot Light e Warm Standby può talvolta essere difficile da comprendere. Entrambe prevedono un ambiente nella tua regione di ripristino con copie degli asset della tua regione principale. La differenza è che la strategia Pilot Light non può elaborare le richieste senza aver prima intrapreso altre azioni, mentre Warm Standby può gestire immediatamente il traffico (a livelli ridotti di capacità). La strategia Pilot Light richiede l'attivazione dei server, possibilmente l'implementazione di un'infrastruttura aggiuntiva (non principale) e l'aumento di risorse, mentre Warm Standby richiede solo l'aumento di risorse (tutto è già stato implementato ed è in esecuzione). Scegli tra queste opzioni in base alle tue esigenze di RTO e RPO.
>
> Quando i costi sono un motivo di preoccupazione e vuoi realizzare obiettivi RPO ed RTO simili a quelli definiti nella strategia di Warm Standby, puoi prendere in considerazione soluzioni native nel cloud, come AWS Elastic Disaster Recovery, che adotta l'approccio Pilot Light e offre obiettivi RPO ed RTO migliori.

Passaggi dell'implementazione

1. Definisci una strategia di ripristino di emergenza in linea con i requisiti di ripristino di questo carico di lavoro.

La scelta di una strategia di ripristino di emergenza è un compromesso tra la riduzione dei tempi di inattività e della perdita di dati (RTO ed RPO) e i costi e la complessità di implementazione della strategia. Dovresti evitare di implementare una strategia che sia più severa del necessario, in quanto questo comporterebbe costi aggiuntivi.

Ad esempio, nel diagramma seguente, l'azienda ha stabilito l'RTO massimo concesso e il limite di spesa per la strategia di ripristino del servizio. Considerati gli obiettivi dell'azienda, le strategie di ripristino di emergenza Pilot Light o di Warm Standby soddisfano sia l'RTO sia i criteri per i costi.

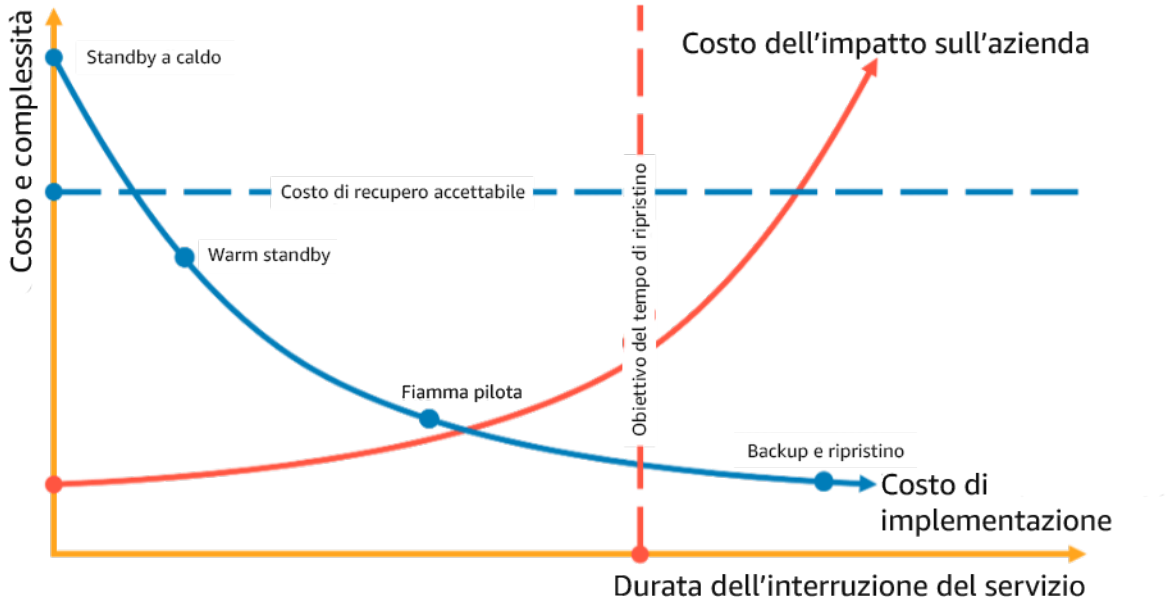


Figure 18: Scegliere una strategia di ripristino di emergenza in base all'RTO e ai costi

Per ulteriori informazioni, consulta Piano di continuità aziendale.

2. Esamina i modelli con cui la strategia di ripristino di emergenza selezionata può essere implementata.

Questo passaggio consiste nel capire come implementare la strategia selezionata. Le strategie vengono spiegate con Regioni AWS come siti principali e di ripristino. Tuttavia, puoi anche decidere di utilizzare le zone di disponibilità in una singola regione come strategia di ripristino di emergenza, utilizzando aspetti di più strategie.

Nei passaggi seguenti puoi applicare la strategia al carico di lavoro specifico.

Backup e ripristino

La strategia di backup e ripristino è la meno complessa da implementare, ma richiede più tempo e impegno per il ripristino del carico di lavoro, causando un RTO e un RPO più elevati. È buona pratica creare sempre backup dei dati e copiarli in un altro sito (ad esempio, un altro Regione AWS).

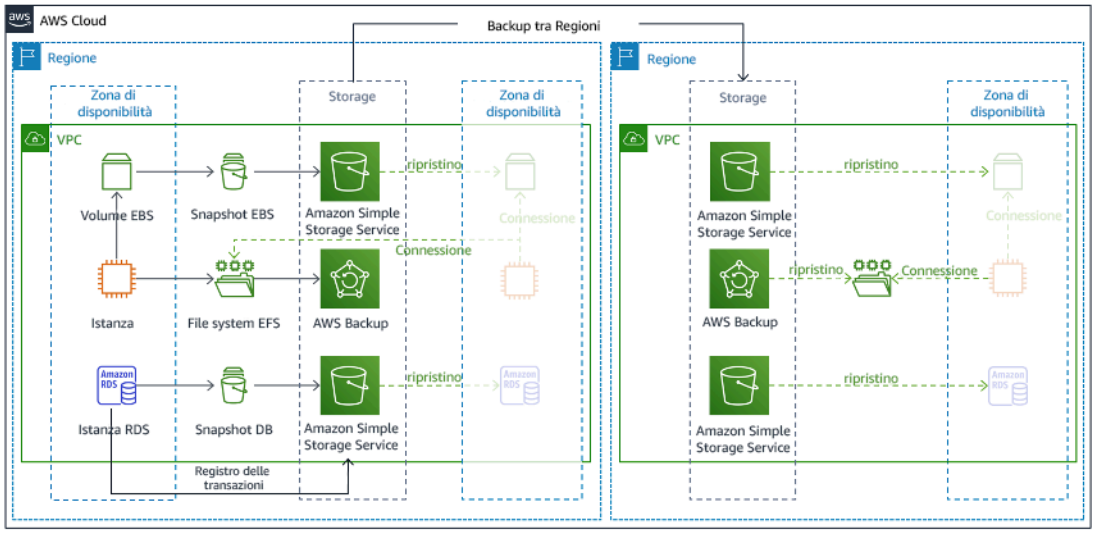


Figura 19: architettura di backup e ripristino

Per ulteriori informazioni su questa strategia, consulta [Architettura di ripristino di emergenza su AWS, parte II: backup e ripristino con recupero rapido].

Pilot light

Con l'approccio Pilot Light puoi replicare i dati dalla regione primaria alla regione di ripristino. Le risorse di base utilizzate per l'infrastruttura del carico di lavoro vengono distribuite nella regione di ripristino; tuttavia sono comunque necessarie risorse aggiuntive ed eventuali dipendenze per rendere funzionale questo stack. Ad esempio, nella Figura 20 non viene implementata alcuna risorsa di calcolo.

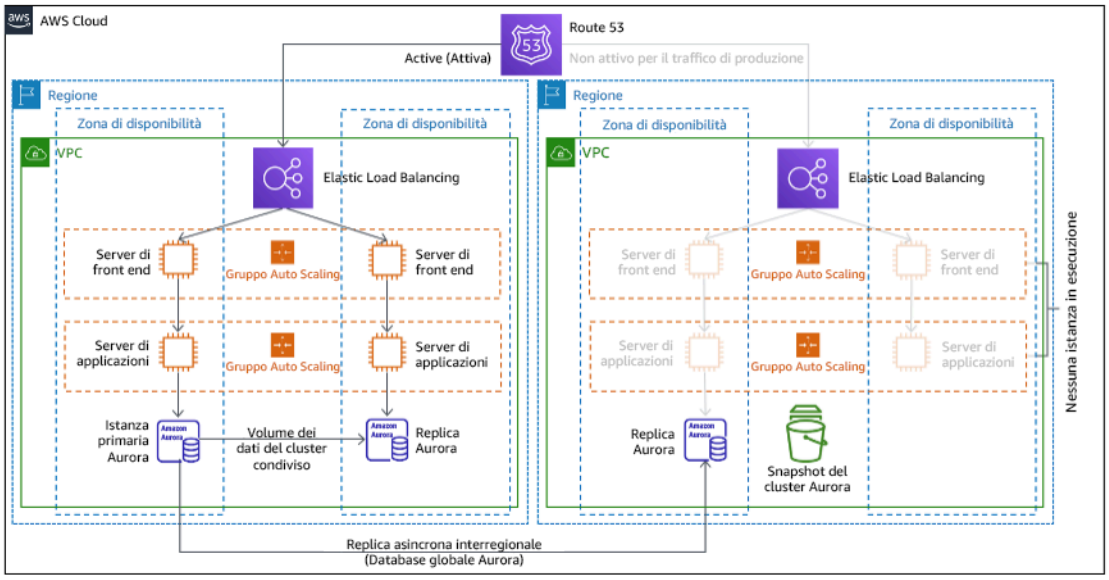


Figura 20: architettura pilot light

Per ulteriori informazioni su questa strategia, consulta [Architettura di ripristino di emergenza su AWS, parte III: Pilot Light e Warm Standby].

Warm standby

L'approccio Warm Standby garantisce che vi sia una copia ridotta ma completamente funzionale dell'ambiente di produzione in un'altra regione. Questo approccio estende il concetto di Pilot Light e diminuisce il tempo di ripristino, poiché il carico di lavoro è sempre attivo in un'altra regione. Se la regione di ripristino è implementata alla massima capacità, la strategia è nota come standby a caldo.

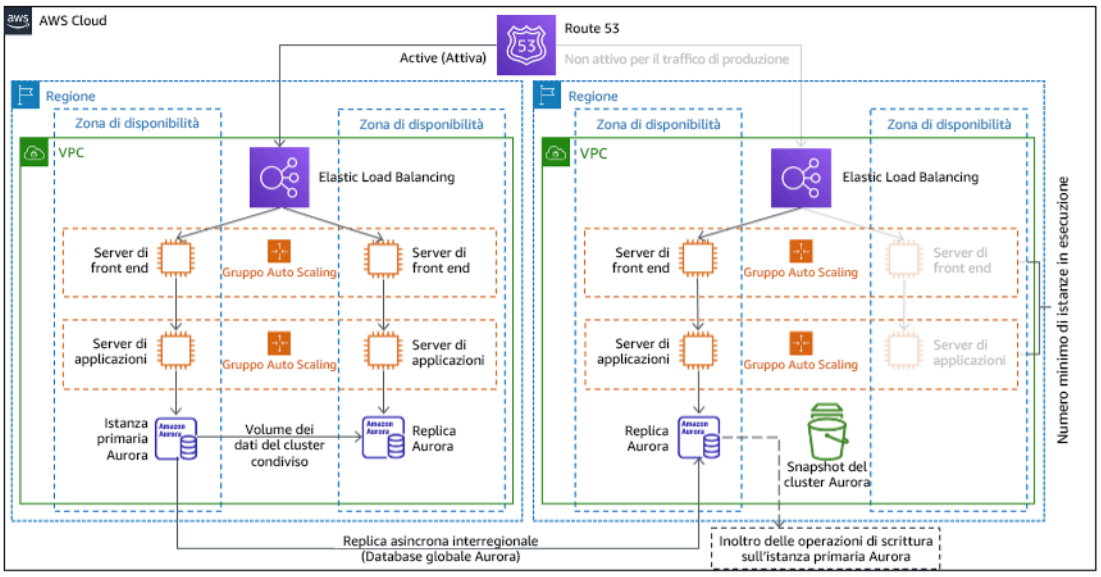


Figure 21: Architettura Warm Standby

Se si utilizza Warm Standby o Pilot Light è necessario un aumento delle risorse nella regione di ripristino. Per verificare che sia disponibile capacità sufficiente quando necessario, valuta se usare prenotazioni di capacità per istanze EC2. Se usi AWS Lambda, la concorrenza assegnata può fornire ambienti di esecuzione pronti a rispondere immediatamente alle chiamate della funzione.

Per ulteriori informazioni su questa strategia, consulta Architettura di ripristino di emergenza su AWS, parte III: Pilot Light e Warm Standby.

Attivo/attivo multi-sito

Puoi eseguire il carico di lavoro simultaneamente in più regioni come parte di una strategia attivo/attivo multisito. La strategia attivo/attivo multi-sito serve il traffico da tutte le regioni in cui è distribuita. I clienti possono selezionare questa strategia per motivi diversi dal ripristino di emergenza. Può essere utilizzata per aumentare la disponibilità o nella distribuzione di un carico di lavoro a un pubblico globale (per posizionare l'endpoint più vicino agli utenti e/o per distribuire stack localizzati al pubblico di quella regione). Come strategia di ripristino di emergenza, se il carico di lavoro non può essere supportato in una delle Regioni AWS in cui viene implementato, la regione viene evacuata e vengono usate le regioni rimanenti per garantire la disponibilità. Attivo/attivo multi-sito è la strategia di ripristino operativamente più complessa e dovrebbe essere selezionata solo quando lo richiedono i requisiti aziendali.

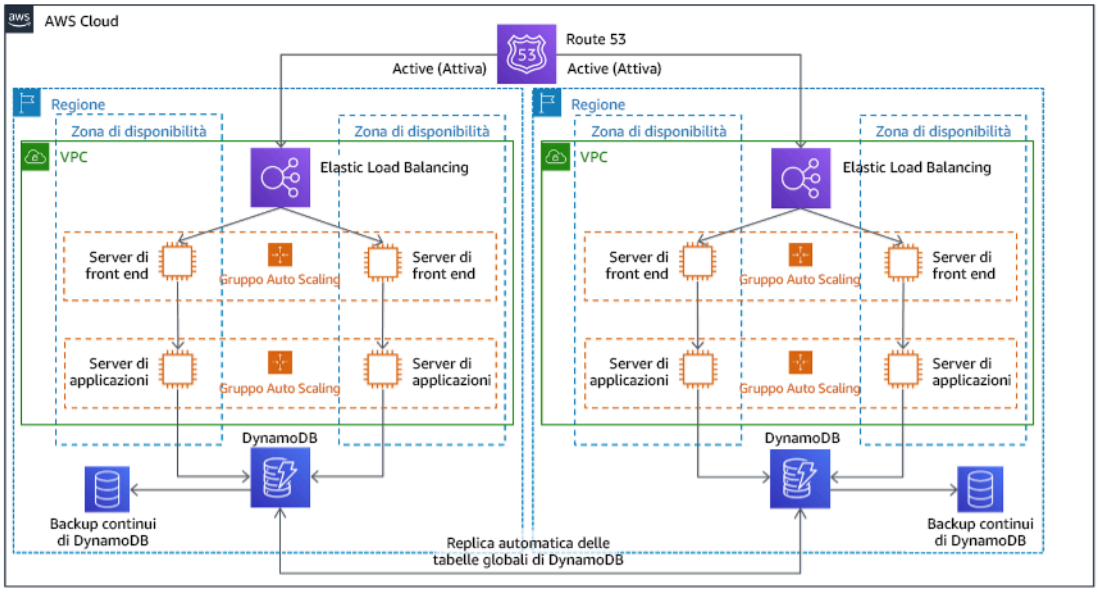


Figura 22: Architettura attivo/attivo multi-sito

Per ulteriori informazioni su questa strategia, consulta Architettura di ripristino di emergenza su AWS, parte IV: attivo/attivo multisito.

AWS Elastic Disaster Recovery

Se stai prendendo in considerazione la strategia Pilot Light o di Warm Standby per il ripristino di emergenza, AWS Elastic Disaster Recovery può fornire un approccio alternativo con vantaggi ancora migliori. Elastic Disaster Recovery può offrire obiettivi RPO e RTO simili al Warm Standby, ma mantenendo l'approccio a basso costo della strategia Pilot Light. Elastic Disaster Recovery replica i dati dalla regione primaria alla regione di ripristino, usando una protezione continua dei dati per realizzare un RPO misurato in secondi e un RTO che può essere misurato in minuti. Solo le risorse necessarie per replicare i dati vengono implementate nella regione di ripristino, mantenendo i costi ridotti come nella strategia Pilot Light. Quando usi Elastic Disaster Recovery, il servizio coordina e orchestra il ripristino delle risorse di calcolo quando viene avviato come parte di un failover o di un'esercitazione.

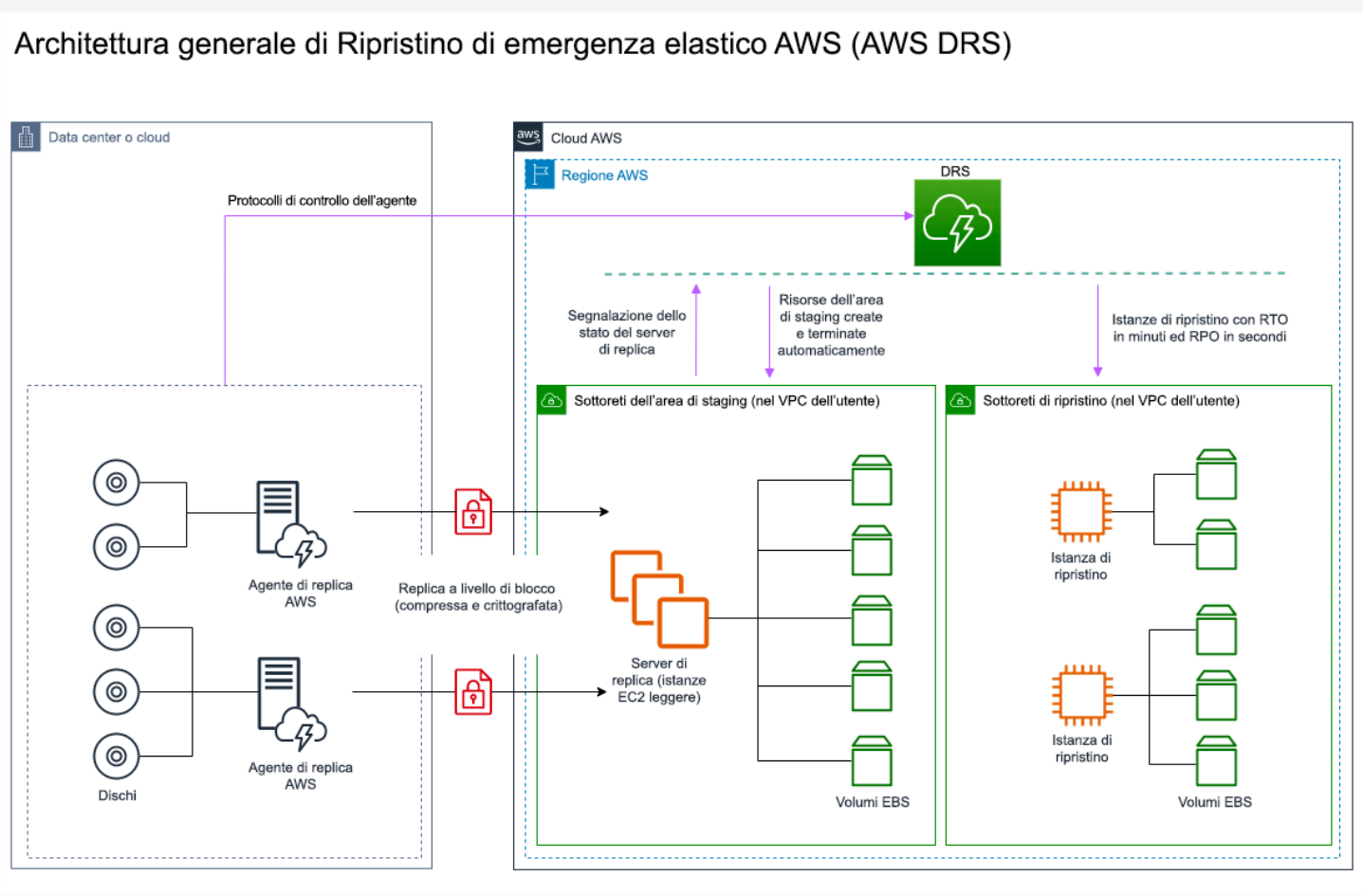


Figura 23: Architettura di AWS Elastic Disaster Recovery

Procedure aggiuntive per la protezione dei dati

Con tutte le strategie devi anche mitigare un disastro relativo ai dati. La replica continua dei dati ti proteggerà da alcuni tipi di disastri, ma non dalla corruzione o dalla distruzione dei dati, a meno che la tua soluzione non includa opzioni per il ripristino point-in-time o il controllo delle versioni dei dati archiviati. Devi anche creare un backup dei dati replicati nel sito di ripristino per creare backup point-in-time in aggiunta alle repliche.

Uso di più zone di disponibilità in una singola Regione AWS

Quando si usano più zone di disponibilità all'interno di un'unica regione, l'implementazione della strategia di ripristino di emergenza usa più elementi delle strategie precedenti. Devi innanzitutto creare un'architettura con disponibilità elevata usando più zone di disponibilità, come mostrato nella Figura 23. Questa architettura usa un approccio attivo/attivo multisito, in quanto le istanze Amazon EC2 e l'Elastic Load Balancer hanno risorse implementate in più zone di disponibilità per la gestione attiva delle richieste. L'architettura presenta anche la strategia di standby a caldo, in cui se l'istanza Amazon RDS primaria (o la zona di disponibilità stessa) restituisce un errore, l'istanza in standby viene promossa a istanza primaria.

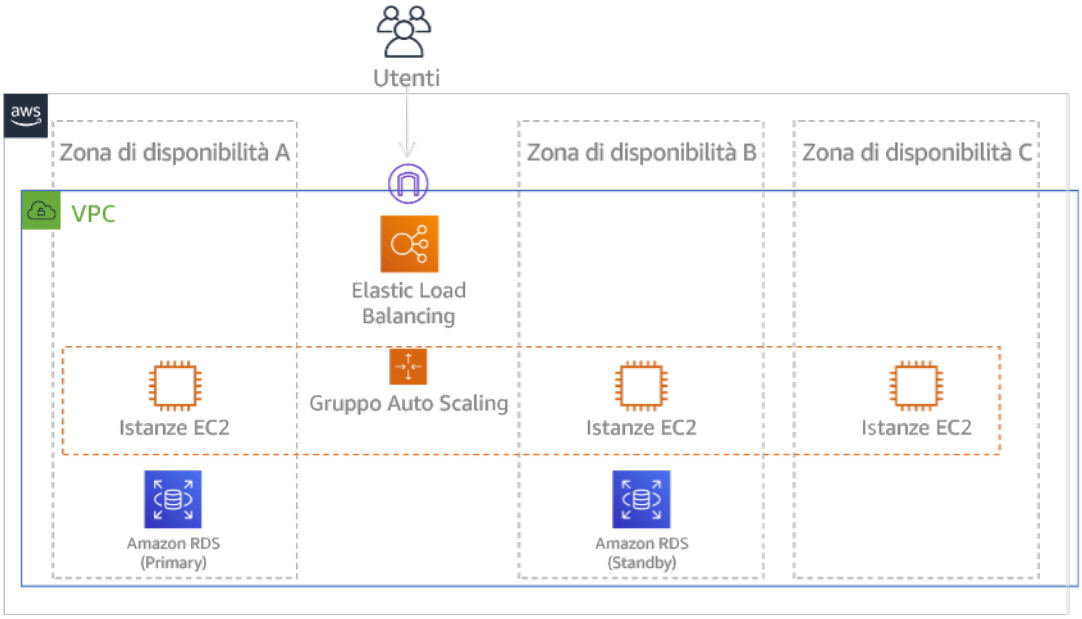


Figura 24: Architettura multi-AZ

Oltre a questa architettura HA, devi aggiungere i backup di tutti i dati richiesti per eseguire il tuo carico di lavoro. Questo aspetto è particolarmente importante per i dati limitati a un'unica zona come i volumi Amazon EBS o i cluster Amazon Redshift. Se fallisce una zona di disponibilità, dovrai ripristinare i dati in un'altra zona di disponibilità. Laddove possibile, devi anche copiare i backup di dati su un'altra Regione AWS come livello di protezione aggiuntivo.

Un approccio alternativo meno comune alla singola regione, ovvero il ripristino di emergenza multi-AZ, viene descritto nel post di blog Creazione di applicazioni altamente resilienti usando il Sistema di controllo Amazon Route 53 per il ripristino di applicazioni, parte 1: stack a regione singola. In questo caso la strategia adottata è quella di garantire il più possibile l'isolamento tra le zone di disponibilità, ossia come le regioni operano. Usando questa strategia alternativa puoi scegliere un approccio attivo/attivo o attivo/passivo.

> **Note**
>
> Alcuni carichi di lavoro hanno requisiti normativi di residenza dei dati. Se questo si applica a un carico di lavoro in una località che attualmente ha solo una Regione AWS, la multi-regione non soddisferà i requisiti aziendali. Le strategie con più zone di disponibilità offrono una buona protezione dalla maggior parte dei disastri.

3. Valuta le risorse del tuo carico di lavoro e quale sarà la loro configurazione nella regione di ripristino prima del failover (durante la normale operatività).

Per l'infrastruttura e le risorse AWS, usa una soluzione Infrastruttura come codice (IaC), come AWS CloudFormation o strumenti di terze parti come Hashicorp Terraform. Per un'implementazione tra più account e regioni con un'unica operazione, puoi usare AWS CloudFormation StackSets. Per le strategie multi-sito attivo/attivo e standby a caldo, l'infrastruttura distribuita nella tua regione di ripristino ha le stesse risorse della regione principale. Per le strategie Pilot Light e Warm Standby l'infrastruttura distribuita richiederà azioni aggiuntive per essere pronta per la produzione. Usando parametri e logica condizionale in CloudFormation, puoi controllare se uno stack implementato sia attivo o in standby con un unico modello. Quando usi Elastic Disaster Recovery, il servizio replica e orchestra il ripristino delle configurazioni delle applicazioni e delle risorse di calcolo.

Tutte le strategie di ripristino di emergenza richiedono il backup delle origini dati all'interno della Regione AWS e i backup vengono quindi copiati nella regione di ripristino. AWS Backup offre una visualizzazione centralizzata in cui puoi configurare, pianificare e monitorare i backup per queste risorse. Per gli approcci Pilot Light, di Warm Standby e attivo/attivo multisito, devi anche replicare i dati dalla regione primaria alle risorse di dati nella regione di ripristino, come Amazon Relational Database Service (Amazon RDS), istanze di database o tabelle Amazon DynamoDB. Queste risorse di dati sono pertanto attive e pronte per servire le richieste nella regione di ripristino.

Per ulteriori informazioni sul funzionamento dei servizi AWS tra regioni, consulta questa serie di blog sulla creazione di un'applicazione in più regioni con servizi AWS.

4. Stabilisci e implementa le modalità con cui preparerai la tua regione al failover nel momento in cui sarà necessario (durante un evento disastroso).

Per la strategia attivo/attivo multisito, il failover significa evacuare una regione e usare le regioni attive rimanenti. In generale, tali regioni sono pronte per accettare il traffico. Per le strategie Pilot Light e di Warm Standby, le azioni di ripristino devono implementare le risorse mancanti, come le istanze EC2 nella Figura 20, insieme a risorse mancanti di altro tipo.

Per tutte le strategie precedenti potresti dover promuovere istanze di database i sola lettura a istanze di lettura/scrittura principali.

Per il backup e il ripristino, il ripristino dei dati dai backup crea risorse per tali dati, come volumi EBS, istanze DB RDS e tabelle DynamoDB. Devi anche ripristinare l'infrastruttura e distribuire il codice. Puoi usare AWS Backup per ripristinare i dati nella regione di ripristino. Consulta REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o riproduzione dei dati dalle origini per ulteriori dettagli. La ricreazione dell'infrastruttura include la creazione di risorse come le istanze EC2, insieme a Amazon Virtual Private Cloud (Amazon VPC), alle sottoreti e ai gruppi di sicurezza necessari. Puoi automatizzare gran parte del processo di ripristino. Per informazioni su come fare, consulta questo post di blog.

5. Stabilisci e implementa le modalità con cui reindirizzerai il traffico al failover nel momento in cui sarà necessario (durante un evento disastroso).

Questa operazione di failover può essere avviata automaticamente o manualmente. Il failover avviato automaticamente in base a controlli dell'integrità o allarmi deve essere usato con attenzione, poiché un failover non necessario (falso allarme) comporta dei costi in termini di non disponibilità e perdita dei dati. Pertanto si usa spesso il failover avviato manualmente. In questo caso, devi comunque automatizzare i passaggi del failover, in modo che l'avvio manuale si limiti al clic su un pulsante.

Esistono diverse opzioni di gestione del traffico da considerare quando si usano i servizi AWS. Un'opzione consiste nell'usare Amazon Route 53. Con Amazon Route 53 puoi associare più endpoint IP in una o più Regioni AWS con un nome di dominio Route 53. Per implementare un failover avviato manualmente, puoi usare il Sistema di controllo Amazon Route 53 per il ripristino di applicazioni, che fornisce un'API del piano dati a disponibilità elevata per reinstradare il traffico nella regione di ripristino. Nella fase di implementazione del failover, usa le operazioni di piano dati ed evita quelle del piano di controllo come descritto in REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo durante il ripristino.

Per ulteriori informazioni su questa e altre opzioni, consulta questa sezione del whitepaper sul ripristino di emergenza.

6. Progetta un piano per il failback del carico di lavoro.

Si parla di failback quando un'operazione del carico di lavoro torna alla regione principale, dopo che un vento disastroso è diminuito di intensità. Il provisioning di infrastruttura e codice alla regione principale in genere segue gli stessi passaggi usati inizialmente, affidandosi all'infrastruttura come codice e alle pipeline di distribuzione del codice. La sfida del failback è il ripristino dei data store e la garanzia della loro coerenza con la regione di ripristino attiva.

Nello stato di failover i database nella regione di ripristino sono attivi e hanno dati aggiornati. L'obiettivo è eseguire una nuova sincronizzazione tra la regione di ripristino e la regione principale, per garantire il suo aggiornamento.

Alcuni servizi AWS eseguono questa operazione in automatico. Se quando usi tabelle globali Amazon DynamoDB la tabella nella regione primaria diventa non disponibile, quando torna online DynamoDB riprende la propagazione delle scritture in sospeso. Se usi il Database globale Amazon Aurora e un failover pianificato gestito, viene mantenuta la topologia di replica esistente del Database globale Aurora. Pertanto, l'istanza precedente in lettura/scrittura nella regione principale diventa una replica e riceve gli aggiornamenti dalla regione di ripristino.

Nei casi in cui questo non è automatico devi ristabilire il database nella regione principale come replica del database nella regione di ripristino. In molti casi questo comporterà l'eliminazione del database principale precedente e la creazione di nuove repliche. Ad esempio, per istruzioni su come fare usando il Database globale Amazon Aurora presupponendo un failover non pianificato, consulta questo lab: Failback di un database globale.

Dopo un failover, se puoi proseguire l'esecuzione nella tua regione di ripristino, valuta la possibilità di farlo nella tua regione principale. Compieresti comunque tutte le operazioni precedenti per trasformare la precedente regione principale in una regione di ripristino. Alcune organizzazioni eseguono una rotazione pianificata, scambiando periodicamente le regioni principale e di ripristino (ad esempio, ogni tre mesi).

Tutti i passaggi richiesti per failover e failback devono essere inseriti in un playbook disponibile a tutti i membri del team, sottoposto periodicamente a revisione.

Quando usi Elastic Disaster Recovery, il servizio fornirà assistenza per l'orchestrazione e l'automazione del processo di failback. Per ulteriori informazioni, consulta Esecuzione di un failback.

Livello di impegno per il piano di implementazione: elevato

Risorse

Best practice correlate:

- the section called “REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o riproduzione dei dati dalle origini”
- the section called “REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo durante il ripristino”
- the section called “REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati”

Documenti correlati:

- AWS Architecture Blog: serie sul ripristino di emergenza
- Ripristino di emergenza dei carichi di lavoro su AWS: ripristino nel cloud (whitepaper AWS)
- Opzioni di ripristino di emergenza nel cloud
- Build a serverless multi-region, active-active backend solution in an hour
- Multi-region serverless backend — reloaded
- RDS: creazione di una replica di lettura in una regione AWS diversa
- Route 53: configurazione del failover DNS
- S3: replica tra regioni
- Che cos'è AWS Backup?
- Che cos'è il Sistema di controllo Route 53 per il ripristino di applicazioni?
- Ripristino di emergenza elastico AWS
- HashiCorp Terraform: Get Started - AWS
- Partner APN: partner che possono assistere con disaster recovery
- Marketplace AWS: prodotti che possono essere usati per il ripristino di emergenza

Video correlati:

- Ripristino di emergenza per carichi di lavoro su AWS
- AWS re:Invent 2018: Modelli architetturali per applicazioni attivo/attivo multi-regione (ARC209-R2)

- Nozioni di base sul ripristino di emergenza elastico AWS | Amazon Web Services

Esempi correlati:

- Well-Architected Lab: Ripristino di emergenza – Serie di workshop che descrivono le strategie di ripristino di emergenza

REL13-BP03 Esecuzione di test sull'implementazione del ripristino di emergenza per convalidare l'implementazione

Testa regolarmente il failover nel sito di ripristino per verificare che funzioni correttamente e che sia possibile soddisfare l'RTO e l'RPO.

Anti-pattern comuni:

- Non eseguire mai failover di prova in produzione.

Vantaggi dell'adozione di questa best practice: l'esecuzione regolare di test del piano di ripristino di emergenza permette di verificare che funzionerà quando necessario e che il team è in grado di eseguire la strategia.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

Un modello da evitare è lo sviluppo di percorsi di ripristino eseguiti raramente. Ad esempio, è possibile che si disponga di un archivio dati secondario utilizzato per query di sola lettura. Quando scrivi in un archivio dati e quello principale ha un guasto, puoi eseguire il failover verso l'archivio dati secondario. Se non testi frequentemente questo failover, è possibile che i presupposti relativi alle funzionalità dell'archivio dati secondario non siano corretti. La capacità dell'archivio dati secondario, che potrebbe essere stata sufficiente durante l'ultimo test, potrebbe non essere più in grado di tollerare il carico in questo scenario. La nostra esperienza ha dimostrato che l'unico ripristino da errore che funziona è il percorso sottoposto a frequenti test. Per questo è preferibile avere un numero ridotto di percorsi di ripristino. Puoi stabilire dei modelli di ripristino e testarli regolarmente. Se disponi di un percorso di ripristino complesso o critico, devi comunque riprodurre regolarmente il guasto specifico in produzione per convincerti che il percorso di ripristino funzioni. Nell'esempio appena discusso, è necessario eseguire il failover regolarmente in standby, indipendentemente dalle necessità.

Passaggi dell'implementazione

1. Progetta i carichi di lavoro per il ripristino. Esegui regolarmente test dei tuoi percorsi di ripristino. Il calcolo orientato al ripristino identifica le caratteristiche nei sistemi che migliorano il ripristino: isolamento e ridondanza, ripristino a livello di sistema dello stato precedente rispetto alle modifiche, capacità di fornire diagnostica, ripristino automatico, progettazione modulare e possibilità di riavvio. Prova il percorso di ripristino per verificare di poter completare il ripristino nel tempo specificato e in base allo stato specificato. Usa i tuoi runbook durante questo ripristino per documentare i problemi e trovare le loro soluzioni prima del test successivo.
2. Per carichi di lavoro basati su Amazon EC2, usa AWS Elastic Disaster Recovery per implementare e avviare istanze di prova per la strategia di ripristino di emergenza. AWS Elastic Disaster Recovery permette di eseguire esercitazioni in modo efficiente, semplificando la preparazione a un evento di failover. Puoi anche avviare spesso le istanze usando Elastic Disaster Recovery per scopi di test ed esercitazione senza reindirizzare il traffico.

Risorse

Documenti correlati:

- Partner APN: partner che possono assistere con disaster recovery
- AWS Architecture Blog: serie sul ripristino di emergenza
- Marketplace AWS: prodotti che possono essere usati per il ripristino di emergenza
- AWS Elastic Disaster Recovery
- Ripristino di emergenza dei carichi di lavoro su AWS: ripristino nel cloud (whitepaper AWS)
- AWS Elastic Disaster Recovery – Preparazione per il failover
- Il progetto di informatica orientata al ripristino Berkeley/Stanford
- Che cos'è il Simulatore di iniezione guasti AWS?

Video correlati:

- AWS re:Invent 2018: Modelli architetturali per applicazioni attivo/attivo multi-regione
- AWS re:Invent 2019: Backup-e ripristino e soluzioni di ripristino di emergenza con AWS

Esempi correlati:

- Well-Architected Lab – Esecuzione di test per la resilienza

REL13-BP04 Gestione della deviazione di configurazione nel sito o nella Regione del ripristino di emergenza

Assicurati che l'infrastruttura, i dati e la configurazione soddisfino le esigenze del sito o nella Regione del ripristino di emergenza. Ad esempio, controlla che le AMI e le quote di servizio siano aggiornate.

AWS Config monitora e registra in modo continuo le configurazioni delle risorse AWS. È in grado di rilevare le deviazioni e attivare AWS Systems Manager Automation per risolverle e attivare allarmi. AWS CloudFormation è inoltre in grado di rilevare le deviazioni negli stack distribuiti.

Anti-pattern comuni:

- Non eseguire aggiornamenti nelle sedi di ripristino, quando esegui modifiche di configurazione o di infrastruttura nelle tue sedi principali.
- Ignorare le limitazioni potenziali (ad esempio le differenze di servizio) nelle sedi di disaster recovery e principali.

Vantaggi dell'adozione di questa best practice: Assicurarsi che l'ambiente di disaster recovery sia coerente con quello esistente garantisce il ripristino completo.

Livello di rischio associato se questa best practice non fosse adottata: Medio

Guida all'implementazione

- Assicurati che le tue pipeline di distribuzione riforniscano sia i siti principali che di backup. Le pipeline per la distribuzione di applicazioni in produzione devono essere distribuite in tutte le posizioni della strategia di disaster recovery specificate, inclusi gli ambienti di sviluppo e test.
- Abilitazione di AWS Config per monitorare le potenziali posizioni di deviazione. Utilizza le regole AWS Config per creare sistemi in grado di applicare le strategie di disaster recovery e generare avvisi quando rilevano una deviazione.
  - Correzione di risorse AWS non conformi in base alle regole di Regole di AWS Config
  - AWS Systems Manager Automation
- Utilizza AWS CloudFormation per distribuire la tua infrastruttura. AWS CloudFormation è in grado di rilevare le deviazioni tra ciò che i modelli di CloudFormation specificano e ciò che viene effettivamente distribuito.
  - AWS CloudFormation: rilevamento delle deviazioni su un intero stack CloudFormation

Risorse

Documenti correlati:

- Partner APN: partner che possono assistere con disaster recovery
- AWS Architecture Blog: Disaster Recovery Series
- AWS CloudFormation: rilevamento delle deviazioni su un intero stack CloudFormation
- Marketplace AWS: prodotti utilizzabili per il disaster recovery
- AWS Systems Manager Automation
- Ripristino di emergenza dei carichi di lavoro su AWS: ripristino nel cloud (whitepaper di AWS)
- In che modo è possibile implementare una soluzione di gestione della configurazione dell'infrastruttura in AWS?
- Correzione di risorse AWS non conformi in base alle regole di Regole di AWS Config

Video correlati:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (Modelli di architettura per applicazioni attive-attive multiregione) (ARC209-R2)

REL13-BP05 Automatizzazione del ripristino

Utilizza AWS o strumenti di terze parti per automatizzare il ripristino del sistema e instradare il traffico verso il sito o la Regione del ripristino di emergenza.

In base ai controlli di integrità configurati, i servizi AWS, come Elastic Load Balancing e AWS Auto Scaling, possono distribuire il carico a zone di disponibilità integre, mentre i servizi, come Amazon Route 53 e AWS Global Accelerator, instradano il carico a Regioni AWS integre. Amazon Route 53 Application Recovery Controller aiuta a gestire e coordinare il failover utilizzando i controlli di disponibilità e le funzionalità di controlli di routing. Queste funzionalità monitorano continuamente la capacità dell'applicazione di riprendersi dai guasti e permettono di controllarne il ripristino delle applicazioni su più Regioni AWS, zone di disponibilità e on-premise.

Per i carichi di lavoro su data center fisici o virtuali o cloud privati, Ripristino di emergenza elastico AWS, disponibile tramite Marketplace AWS, consente alle organizzazioni di organizzare una strategia di ripristino di emergenza su AWS. CloudEndure supporta, inoltre, il ripristino di emergenza tra Regioni e zone di disponibilità in AWS.

Anti-pattern comuni:

- L'implementazione di failover e failback automatici identici può causare flapping quando si verifica un errore.

Vantaggi dell'adozione di questa best practice: Il ripristino automatico riduce i tempi di ripristino eliminando la possibilità di errori manuali.

Livello di rischio associato se questa best practice non fosse adottata: Medio

Guida all'implementazione

- Automatizzazione dei percorsi di ripristino. Per tempi di ripristino brevi, non è possibile servirsi del giudizio umano e dell'azione per scenari di disponibilità elevata. Il sistema dovrebbe ripristinarsi automaticamente in ogni situazione.
  - Usa il ripristino di emergenza CloudEndure per failover e failback automatizzati. Il ripristino di emergenza CloudEndure replica in modo continuo le macchine (tra cui sistema operativo, configurazione dello stato del sistema, database, applicazioni e file) in un'area di gestione temporanea a basso costo nell'Account AWS di destinazione e nella Regione preferita. In caso di emergenza, è possibile indicare a CloudEndure Disaster Recovery di avviare automaticamente migliaia di macchine nello stato di provisioning completo in pochi minuti.
    - Performing a Disaster Recovery Failover and Failback
    - CloudEndure Disaster Recovery

Risorse

Documenti correlati:

- Partner APN: partner che possono assistere con disaster recovery
- AWS Architecture Blog: Disaster Recovery Series
- Marketplace AWS: prodotti utilizzabili per il disaster recovery
- AWS Systems Manager Automation
- Ripristino di emergenza CloudEndure in AWS
- Ripristino di emergenza dei carichi di lavoro su AWS: ripristino nel cloud (whitepaper di AWS)

Video correlati:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (Modelli architetturali per applicazioni attive-attive su più Regioni) (ARC209-R2)

Efficienza delle prestazioni

Il principio dell'efficienza delle prestazioni comprende l'abilità di utilizzare in modo efficiente le risorse di elaborazione per soddisfare i requisiti del sistema e conservare tale efficienza a seconda dei cambiamenti della domanda e dell'evoluzione delle tecnologie. È possibile trovare linee guida prescrittive sull'implementazione nel Whitepaper sul principio dell'efficienza delle prestazioni.

Aree delle best practice

- Scelta dell'architettura
- Elaborazione e hardware
- Gestione dati
- Reti e distribuzione di contenuti
- Processo e cultura

Scelta dell'architettura

Domande

- PERF 1. In che modo selezioni le risorse e l'architettura cloud appropriate per il tuo carico di lavoro?

PERF 1. In che modo selezioni le risorse e l'architettura cloud appropriate per il tuo carico di lavoro?

La soluzione ottimale per un determinato carico di lavoro può variare e le soluzioni spesso combinano molteplici approcci. I carichi di lavoro Well-Architected utilizzano soluzioni multiple e impiegano funzionalità diverse per migliorare le prestazioni.

Best practice

- PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili
- PERF01-BP02 Utilizzo delle indicazioni del provider cloud o di un partner appropriato per conoscere gli schemi di architettura e le best practice

- PERF01-BP03 Influenza dei costi nelle decisioni sull'architettura
- PERF01-BP04 Valutazione dell'influenza dei compromessi sui clienti e sull'efficienza dell'architettura
- PERF01-BP05 Uso delle policy e delle architetture di riferimento
- PERF01-BP06 Uso del benchmarking per guidare le decisioni sull'architettura
- PERF01-BP07 Uso di un approccio basato sui dati per le scelte dell'architettura

PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili

Informati continuamente e identifica i servizi e le configurazioni disponibili che ti aiutano a prendere le decisioni giuste sull'architettura e a migliorare l'efficienza delle prestazioni dei carichi di lavoro.

Anti-pattern comuni:

- Utilizzi il cloud come data center in co-location.
- Non stai modernizzando la tua applicazione con la migrazione al cloud.
- Stai solo usando un tipo di archiviazione per tutte le cose che devono essere conservate in modo persistente.
- Se necessario, utilizzi tipi di istanze strettamente correlate ai tuoi standard attuali, ma più grandi.
- Distribuisci e gestisci le tecnologie disponibili come servizi gestiti.

Vantaggi dell'adozione di questa best practice: Prendendo in considerazione nuovi servizi e configurazioni, puoi migliorare notevolmente le prestazioni, ridurre i costi e ottimizzare le attività necessarie per mantenere il carico di lavoro. Puoi anche accelerare il time-to-value per i prodotti abilitati al cloud.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

AWS rilascia continuamente nuovi servizi e funzionalità in grado di migliorare le prestazioni e ridurre i costi dei carichi di lavoro del cloud. Rimanere aggiornati su questi nuovi servizi e funzionalità è fondamentale per mantenere l'efficacia delle prestazioni nel cloud. La modernizzazione dell'architettura dei carichi di lavoro consente inoltre di accelerare la produttività, promuovere l'innovazione e sbloccare ulteriori opportunità di crescita.

Passaggi dell'implementazione

- Esegui l'inventario del software e dell'architettura del carico di lavoro per i servizi correlati. Determina su quale categoria di prodotti ottenere ulteriori informazioni.
- Esplora le offerte AWS per individuare e conoscere i servizi e le opzioni di configurazione pertinenti che possono aiutarti a migliorare le prestazioni e ridurre i costi e la complessità operativa.
  - Cloud Amazon Web Services
  - Academy AWS
  - Novità di AWS
  - Blog AWS
  - AWS Skill Builder
  - Eventi e webinar AWS
  - AWS Training e certificazioni
  - Canale YouTube di AWS
  - Workshop di AWS
  - Community AWS
- Usa gli ambienti sandbox non di produzione per comprendere e sperimentare nuovi servizi senza incorrere in costi aggiuntivi.
- Scopri servizi e funzionalità cloud sempre nuovi.

Risorse

Documenti correlati:

- Panoramica di Amazon Web Services
- Caratteristiche di Amazon EC2
- Impara passo per passo con il Programma di apprendimento dei Partner AWS
- Formazione e certificazione AWS
- My learning path to become an AWS solutions architect
- Centro di progettazione AWS
- AWS Partner Network
- Portfolio di soluzioni AWS
- Knowledge Center di AWS

- Costruisci applicazioni moderne su AWS

Video correlati:

- AWS re:Invent 2023 - What's new with Amazon EC2
- AWS re:Invent 2022 - Reduce your operational and infrastructure costs with Amazon ECS
- AWS re:Invent 2023 - Build with the efficiency, agility & innovation of the cloud with AWS
- AWS re:Invent 2022 - Deploy ML models for inference at high performance and low cost
- La mia architettura

Esempi correlati:

- Esempi di AWS
- Esempi di SDK AWS

PERF01-BP02 Utilizzo delle indicazioni del provider cloud o di un partner appropriato per conoscere gli schemi di architettura e le best practice

Usa le risorse aziendali del cloud come documentazione, solutions architect, servizi professionali o partner appropriati per guidare le tue decisioni sull'architettura. Queste risorse ti aiutano a rivedere e migliorare l'architettura per ottenere prestazioni ottimali.

Anti-pattern comuni:

- AWS è usato come un comune provider di servizi cloud.
- I servizi AWS vengono utilizzati in modo diverso rispetto alla loro progettazione iniziale.
- Le indicazioni vengono seguite senza considerare il contesto aziendale.

Vantaggi dell'adozione di questa best practice: avvalersi della guida di un provider di servizi cloud o di un partner appropriato può aiutarti a fare le scelte giuste per l'architettura del tuo carico di lavoro e darti fiducia nelle tue decisioni.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

AWS offre un'ampia scelta di linee guida, documentazione e risorse che possono aiutarti a creare e gestire i carichi di lavoro del cloud in modo efficiente. La documentazione AWS fornisce esempi di codice, esercitazioni e spiegazioni dettagliate sui servizi. Oltre alla documentazione, AWS offre programmi di formazione e certificazione, solutions architect e servizi professionali che i clienti possono usare per esplorare diversi aspetti dei servizi cloud e implementare un'architettura cloud efficiente su AWS.

Sfrutta queste risorse per ottenere approfondimenti sulle informazioni e sulle best practice preziose per risparmiare tempo e ottenere risultati migliori nel Cloud AWS.

Passaggi dell'implementazione

- Consulta la documentazione e le linee guida AWS e segui le best practice. Queste risorse possono aiutarti a scegliere e configurare i servizi in modo efficace e a ottenere prestazioni migliori.
  - Documentazione di AWS (come guide utente e white paper)
  - Blog AWS
  - AWS Training e certificazioni
  - Canale YouTube di AWS
- Partecipa agli eventi per i partner AWS (come summit AWS a livello mondiale, gruppi di utenti di AWS re:Invent e workshop) per apprendere dagli esperti AWS le best practice per l'utilizzo dei servizi AWS.
  - Impara passo per passo con il Programma di apprendimento dei Partner AWS
  - Eventi e webinar AWS
  - Workshop di AWS
  - Community AWS
- Contatta AWS per ricevere assistenza quando ti occorrono ulteriori indicazioni o informazioni sui prodotti. Gli AWS Solutions Architect e AWS Professional Services forniscono linee guida per l'implementazione della soluzione. I Partner AWS mettono a disposizione la propria competenza su AWS per aiutarti ad assicurare alla tua azienda agilità ed innovazione.
- utilizza AWS Support se hai bisogno di supporto tecnico per utilizzare un servizio in modo efficace. I nostri piani di assistenza sono pensati per offrirti il giusto mix di strumenti e competenze in modo da poter conseguire il successo con AWS ottimizzando le prestazioni, gestendo i rischi e tenendo sotto controllo i costi.

Risorse

Documenti correlati:

- Centro di progettazione AWS
- AWS Partner Network
- Portfolio di soluzioni AWS
- Knowledge Center di AWS
- Supporto Enterprise di AWS

Video correlati:

- La mia architettura
- AWS re:Invent 2023 - Advanced event-driven patterns with Amazon EventBridge
- AWS re:Invent 2023 - Implementing distributed design patterns on AWS
- AWS re:Invent 2023 - Application architecture as code

Esempi correlati:

- Esempi di AWS
- Esempi di SDK AWS
- AWS Analytics Reference Architecture

PERF01-BP03 Influenza dei costi nelle decisioni sull'architettura

Tieni conto dei costi nelle decisioni sull'architettura per migliorare l'utilizzo delle risorse e l'efficienza delle prestazioni del tuo carico di lavoro cloud. Quando si è consapevoli delle implicazioni dei costi del carico di lavoro cloud, è più probabile che si utilizzino risorse efficienti e si riducano le procedure inutili.

Anti-pattern comuni:

- Utilizzi una sola famiglia di istanze.
- Ometti di valutare le soluzioni con licenza rispetto alle soluzioni open-source.
- Non definisci le policy del ciclo di vita dell'archiviazione.

- Non esamini i nuovi servizi e funzionalità del Cloud AWS.
- Utilizzi solo lo storage a blocchi.

Vantaggi dell'adozione di questa best practice: La contabilizzazione dei costi nel processo decisionale consente di utilizzare risorse più efficienti ed esplorare altri investimenti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

L'ottimizzazione dei carichi di lavoro in base ai costi può migliorare l'utilizzo delle risorse ed evitare sprechi nel carico di lavoro cloud. Tenere conto dei costi nelle decisioni sull'architettura di solito include il corretto dimensionamento dei componenti del carico di lavoro e l'abilitazione dell'elasticità, comportando una migliore efficienza delle prestazioni del carico di lavoro cloud.

Passaggi dell'implementazione

- Stabilisci gli obiettivi di costo, come i limiti del budget, per il tuo carico di lavoro cloud.
- Identifica i componenti chiave, come istanze e archiviazione, che determinano il costo del carico di lavoro. Puoi utilizzare AWS Pricing Calculator e AWS Cost Explorer per identificare i principali fattori di costo del carico di lavoro.
- Comprensione dei modelli di prezzo nel cloud, ad esempio istanze on-demand, riservate, Savings Plans e spot.
- Utilizza Migliori pratiche di ottimizzazione dei costi di Well-Architected per ottimizzare questi componenti chiave in termini di costi.
- Monitora e analizza continuamente i costi per identificare le opportunità di ottimizzazione dei costi nel tuo carico di lavoro.
  - utilizza Budget AWS per ricevere gli avvisi per i costi inaccettabili.
  - utilizza AWS Compute Optimizer oppure AWS Trusted Advisor per ottenere suggerimenti sull'ottimizzazione dei costi.
  - utilizza Rilevamento delle anomalie dei costi AWS per rilevare in modo automatico le anomalie dei costi e analizzare la causa principale.

Risorse

Documenti correlati:

- Che cos'è la Gestione costi e fatturazione AWS?
- Ottimizzazione dei costi con AWS
- Choosing an AWS cost management strategy
- A Beginner’s Guide to AWS Cost Management
- A Detailed Overview of the Cost Intelligence Dashboard
- Centro di progettazione AWS
- Portfolio di soluzioni AWS
- Knowledge Center di AWS

Video correlati:

- La mia architettura
- AWS re:Invent 2023 - What’s new with AWS cost optimization
- AWS re:Invent 2023 - Optimize cost and performance and track progress toward mitigation
- AWS re:Invent 2023 - AWS storage cost-optimization best practices
- AWS re:Invent 2023 - Optimize costs in your multi-account environments

Esempi correlati:

- AWS Compute Optimizer Demo code (Codice dimostrativo di AWS Compute Optimizer)
- Cost Optimization Workshop
- Cloud Financial Management Technical Implementation Playbooks
- Startup optimization: Tuning application performance for maximum efficiency
- Serverless Optimization Workshop (Performance and Cost)
- Scaling cost effective architectures

PERF01-BP04 Valutazione dell'influenza dei compromessi sui clienti e sull'efficienza dell'architettura

Quando valuti i miglioramenti correlati alle prestazioni, determina quali scelte hanno impatto sui clienti e sull'efficienza del carico di lavoro. Ad esempio, se l'utilizzo di un datastore chiave-valore aumenta le prestazioni del sistema, è importante valutare in che modo la consistenza finale intrinseca di questo cambiamento avrà un impatto sui clienti.

Anti-pattern comuni:

- Ritieni che tutti i vantaggi prestazionali debbano essere implementati, anche se ci sono compromessi per l'implementazione.
- Valuti di apportare modifiche ai carichi di lavoro solo quando un problema prestazionale ha raggiunto un punto critico.

Vantaggi dell'adozione di questa best practice: Quando si valutano potenziali miglioramenti relativi alle prestazioni, è necessario decidere se i compromessi per le modifiche sono accettabili con i requisiti del carico di lavoro. In alcuni casi, potrebbe essere necessario implementare controlli aggiuntivi per compensare i compromessi.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Identifica le aree critiche della tua architettura in termini di prestazioni e impatto sui clienti. Stabilisci in che modo puoi apportare miglioramenti e quali compromessi comportano, oltre al loro impatto sul sistema e sull'esperienza degli utenti. L'implementazione di cache di dati, ad esempio, può contribuire a migliorare notevolmente le prestazioni ma richiede una strategia ben definita sulle modalità e sui tempi di aggiornamento o di invalidamento dei dati che vi sono contenuti, per evitare che il sistema si comporti in modo non corretto.

Passaggi dell'implementazione

- Comprendi i requisiti del tuo carico di lavoro e gli accordi sul livello di servizio (SLA).
- Definisci chiaramente i fattori di valutazione. I fattori possono riguardare il costo, l'affidabilità, la sicurezza e le prestazioni del carico di lavoro.
- Seleziona l'architettura e i servizi in grado di soddisfare le tue esigenze.
- Conduci sperimentazioni e proof of concept (POC) per valutare i fattori di compromesso, l'impatto sui clienti e l'efficienza dell'architettura. Di solito, i carichi di lavoro altamente disponibili, performanti e sicuri consumano più risorse cloud offrendo al contempo una esperienza cliente migliore. Comprendi i compromessi in termini di complessità, prestazioni e costi del tuo carico di lavoro. In genere, dare la priorità a due fattori va a scapito del terzo.

Risorse

Documenti correlati:

- Amazon Builders' Library
- Amazon QuickSight KPIs (KPI di Amazon QuickSight)
- RUM Amazon CloudWatch
- X-Ray Documentation (Documentazione di X-Ray)
- Understand resiliency patterns and trade-offs to architect efficiently in the cloud

Video correlati:

- Optimize applications through Amazon CloudWatch RUM (Ottimizzazione delle applicazioni tramite RUM Amazon CloudWatch)
- AWS re:Invent 2023 - Capacity, availability, cost efficiency: Pick three
- AWS re:Invent 2023 - Advanced integration patterns & trade-offs for loosely coupled systems

Esempi correlati:

- Measure page load time with Amazon CloudWatch Synthetics (Misurare il tempo di caricamento della pagina con Amazon CloudWatch Synthetics)
- Amazon CloudWatch RUM Web Client (Client Web RUM Amazon CloudWatch)

PERF01-BP05 Uso delle policy e delle architetture di riferimento

Utilizza le policy interne e le architetture di riferimento esistenti per la selezione dei servizi e delle configurazioni per essere più efficiente nella progettazione e nell'implementazione del carico di lavoro.

Anti-pattern comuni:

- Usi una vasta gamma di tecnologie che possono influire sul sovraccarico di gestione della tua azienda.

Vantaggi dell'adozione di questa best practice: La definizione di una policy per la scelta dell'architettura, della tecnologia e del fornitore consentirà di prendere decisioni rapidamente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Avere politiche interne nella selezione delle risorse e dell'architettura fornisce standard e linee guida da seguire quando si effettuano scelte architettoniche. Queste linee guida semplificano il processo decisionale nella scelta del servizio cloud giusto e possono contribuire a migliorare l'efficienza delle prestazioni. Distribuisci il carico di lavoro utilizzando policy o architetture di riferimento. Integra i servizi nell'implementazione cloud, quindi utilizza i test delle prestazioni per verificare che i requisiti prestazionali siano sempre rispettati.

Passaggi dell'implementazione

- Comprendi chiaramente i requisiti del tuo carico di lavoro cloud.
- Rivedi le policy interne ed esterne per identificare quelle più pertinenti.
- Utilizza le architetture di riferimento appropriate fornite dalle best practice AWS o di settore.
- Crea un contesto composto da policy, standard, architetture di riferimento e linee guida prescrittive per situazioni comuni. In questo modo i tuoi team possono muoversi più velocemente. Personalizza le risorse per il tuo settore verticale, se applicabile.
- Convalida queste policy e architetture di riferimento per il tuo carico di lavoro in ambienti di sperimentazione (sandbox).
- Rimani aggiornato con gli standard e gli aggiornamenti AWS del settore per assicurarti che le tue policy e le architetture di riferimento ottimizzino il carico di lavoro cloud.

Risorse

Documenti correlati:

- Centro di progettazione AWS
- AWS Partner Network
- Portfolio di soluzioni AWS
- Knowledge Center di AWS
- AWS Architecture Blog

Video correlati:

- La mia architettura
- AWS re:Invent 2022 - Accelerate value for your business with SAP & AWS reference architecture

Esempi correlati:

- Esempi di AWS
- Esempi di SDK AWS

PERF01-BP06 Uso del benchmarking per guidare le decisioni sull'architettura

Esegui il benchmark delle prestazioni di un carico di lavoro esistente per comprendere le prestazioni sul cloud e guidare le decisioni sull'architettura basate sui dati.

Anti-pattern comuni:

- Fai affidamento su valori di riferimento comuni che non sono indicativi delle caratteristiche del carico di lavoro.
- L'unico punto di riferimento è dato dal feedback e dalle percezioni dei clienti.

Vantaggi dell'adozione di questa best practice: il benchmarking dell'attuale implementazione consente di misurare i miglioramenti delle prestazioni.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Utilizza test sintetici di benchmarking per valutare le prestazioni dei componenti durante il carico di lavoro. Di solito, i benchmark sono più rapidi da configurare rispetto ai test di carico e vengono utilizzati per valutare la tecnologia di un componente specifico. Il benchmarking viene spesso utilizzato all'inizio di un nuovo progetto, quando non è ancora disponibile una soluzione completa da sottoporre a test di carico.

Puoi creare test di benchmarking personalizzati oppure utilizzare i test standard del settore, come TPC-DS, per effettuare un'analisi comparativa dei carichi di lavoro. I benchmark di settore sono utili quando devi confrontare ambienti diversi. Quelli personalizzati, invece, sono indicati per analizzare tipi specifici di operazioni che prevedi di eseguire nell'architettura.

In fase di benchmarking, è importante effettuare delle operazioni preliminari sull'ambiente di test al fine di garantire la validità dei risultati. Dovrai eseguire lo stesso benchmark più volte, per verificare di avere acquisito ogni eventuale variazione nel corso del tempo.

Dal momento che, di solito, l'esecuzione dei benchmark è più rapida di quella dei test di carico, il benchmarking può essere utilizzato sin dalle prime fasi della pipeline di distribuzione, così da

fornire al team feedback più rapidi sulle deviazioni delle prestazioni. Quando valuti un cambiamento significativo in un componente o servizio, i benchmark possono essere un modo rapido per verificare se l'impegno necessario per apportare la modifica sia giustificato. L'utilizzo del benchmarking in combinazione con i test di carico è importante perché questi ultimi forniscono indicazioni sulle prestazioni del carico di lavoro in fase di produzione.

Passaggi dell'implementazione

- Pianifica e definisci:
  - Definisci gli obiettivi, la baseline, gli scenari di test, le metriche, ad esempio l'utilizzo della CPU, la latenza o il throughput, e i KPI per il tuo benchmark.
  - Concentrati sui requisiti degli utenti in termini di esperienza utente e su fattori come i tempi di risposta e l'accessibilità.
  - Individua uno strumento di benchmark adatto al tuo carico di lavoro. Puoi utilizzare i servizi AWS, come Amazon CloudWatch, o uno strumento di terze parti compatibile con il carico di lavoro.
- Configura ed esegui l'instrumentazione:
  - Imposta il tuo ambiente e configura le risorse.
  - Implementa il monitoraggio e la registrazione per acquisire i risultati dei test.
- Esegui i test di benchmark e monitora:
  - Esegui i test di benchmark e monitora i parametri durante il test.
- Analizza e documenta:
  - Documenta il processo di benchmark e gli esiti.
  - Analizza i risultati per identificare i colli di bottiglia, le tendenze e le aree di miglioramento.
  - Usa i risultati dei test per prendere decisioni sull'architettura e modificare il carico di lavoro. Questa operazione può includere la modifica dei servizi o l'adozione di nuove funzionalità.
- Ottimizza e ripeti:
  - Modifica le configurazioni e le allocazioni delle risorse in base ai tuoi benchmark.
  - Ripeti il test del carico di lavoro dopo i cambiamenti per convalidare i miglioramenti.
  - Documenta le informazioni e ripeti il processo per identificare altre aree di miglioramento.

Risorse

Documenti correlati:

- Centro di architettura AWS

- AWS Partner Network
- Biblioteca di soluzioni AWS
- Centro conoscenze AWS
- Amazon CloudWatch RUM
- Amazon CloudWatch Synthetics
- Genomics workflows, Part 5: automated benchmarking
- Benchmark and optimize endpoint deployment in Amazon SageMaker JumpStart

Video correlati:

- AWS re:Invent 2023 - Benchmarking AWS Lambda cold starts
- Benchmarking stateful services in the cloud
- La mia architettura
- Optimize applications through Amazon CloudWatch RUM
- Demo di Amazon CloudWatch Synthetics

Esempi correlati:

- Esempi AWS
- Esempi di SDKAWS
- Test di carico distribuito
- Measure page load time with Amazon CloudWatch Synthetics
- Amazon CloudWatch RUM Web Client

PERF01-BP07 Uso di un approccio basato sui dati per le scelte dell'architettura

Definisci un approccio chiaro e basato sui dati per le scelte dell'architettura e verificare che vengano utilizzati i servizi e le configurazioni cloud corretti per soddisfare le tue esigenze aziendali specifiche.

Anti-pattern comuni:

- Ritieni che l'architettura corrente diventi statica e non venga aggiornata nel corso del tempo.
- Le tue scelte dell'architettura si basano su ipotesi e presupposizioni.
- Introduci modifiche all'architettura nel tempo senza giustificazioni.

Vantaggi dell'adozione di questa best practice: Con un approccio ben definito per le scelte dell'architettura, utilizzi i dati per influenzare la progettazione del carico di lavoro e prendere decisioni informate nel tempo.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Affidati all'esperienza e alle competenze interne in materia di cloud o utilizza risorse esterne, come casi d'uso pubblicati o whitepaper, per scegliere risorse e servizi per la tua architettura. È necessario definire con cura un processo che incoraggi la sperimentazione e il benchmarking con i servizi che possono essere utilizzati nel carico di lavoro.

I backlog dei carichi di lavoro critici devono consistere non solo in storie che offrono funzionalità rilevanti per l'azienda e gli utenti, ma anche in storie tecniche che definiscono la presentazione dell'architettura per il carico di lavoro. Questa presentazione include i nuovi progressi tecnologici e i nuovi servizi e li adotta sulla base di dati e giustificazioni adeguate. Verifica che l'architettura sia a prova di futuro e non diventi obsoleta.

Passaggi dell'implementazione

- Interagisci con i principali stakeholder per definire i requisiti del carico di lavoro, comprese le prestazioni, la disponibilità e le considerazioni sui costi. Includi fattori quali il numero di utenti e il modello di utilizzo del tuo carico di lavoro.
- Crea una presentazione dell'architettura o un backlog tecnologico a cui venga assegnata la priorità insieme al backlog funzionale.
- Valuta e identifica i diversi servizi cloud (per maggiori dettagli, consulta PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili).
- Esplora i diversi modelli di architettura, come microservizi o serverless, che soddisfano i tuoi requisiti di prestazioni (per maggiori dettagli, consulta PERF01-BP02 Utilizzo delle indicazioni del provider cloud o di un partner appropriato per conoscere gli schemi di architettura e le best practice).

- Consulta altri team, diagrammi di architettura e risorse, come AWS Solution Architects, Centro di progettazione AWSe AWS Partner Network, per aiutarti a scegliere l'architettura giusta per il tuo carico di lavoro.

- Definisci i parametri, come la velocità di trasmissione effettiva e il tempo di risposta, che possono aiutarti a valutare le prestazioni del tuo carico di lavoro.
- Sperimenta e utilizza i parametri definiti per convalidare le prestazioni dell'architettura selezionata.
- Monitora continuamente e apporta le modifiche necessarie per mantenere ottimali le prestazioni della tua architettura.
- Documenta l'architettura e le decisioni selezionate come riferimento per aggiornamenti e apprendimenti futuri.
- Rivedi e aggiorna continuamente l'approccio di selezione dell'architettura in base agli apprendimenti, alle nuove tecnologie e ai parametri che indicano un problema o un cambiamento necessario nell'approccio attuale.

Risorse

Documenti correlati:

- Portfolio di soluzioni AWS
- Knowledge Center di AWS
- Architectural Patterns to Build End-to-End Data Driven Applications on AWS

Video correlati:

- La mia architettura
- AWS re:Invent 2021 - Data-driven enterprise: Going from vision to value
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2023 - Optimize cost and performance and track progress toward mitigation
- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results

Esempi correlati:

- Esempi di AWS
- Esempi di SDK AWS

Elaborazione e hardware

PERF 2. In che modo selezioni e utilizzi le risorse di elaborazione nel tuo carico di lavoro?

La soluzione ottimale di elaborazione per un determinato sistema potrebbe variare in base alla progettazione dell'applicazione, ai modelli di utilizzo e alle impostazioni di configurazione. Le architetture possono utilizzare diverse soluzioni di calcolo per vari componenti e impiegare funzionalità diverse per migliorare le prestazioni. Selezionare la soluzione di elaborazione sbagliata per un'architettura può ridurre l'efficienza delle prestazioni.

Best practice

- PERF02-BP01 Selezione delle migliori opzioni di elaborazione per il carico di lavoro
- PERF02-BP02 Identificazione delle funzionalità e configurazione di calcolo disponibili
- PERF02-BP03 Raccolta dei parametri relativi al calcolo
- PERF02-BP04 Configurazione e dimensionamento corretto delle risorse di elaborazione
- PERF02-BP05 Dimensionamento dinamico delle risorse di elaborazione
- PERF02-BP06 Uso di acceleratori di elaborazione ottimizzati basati su hardware

PERF02-BP01 Selezione delle migliori opzioni di elaborazione per il carico di lavoro

La selezione dell'opzione di elaborazione più appropriata per il carico di lavoro consente di migliorare le prestazioni, ridurre i costi non necessari dell'infrastruttura e diminuire le attività operative richieste per mantenere il carico di lavoro.

Anti-pattern comuni:

- Si utilizza la stessa opzione di elaborazione utilizzata in locale.
- Non si conoscono le opzioni, le funzionalità e le soluzioni di cloud computing e come queste migliorino le prestazioni di elaborazione.
- Si dimensiona in eccesso l'opzione di elaborazione per soddisfare i requisiti di dimensionamento o prestazioni, quando il passaggio a una nuova opzione di elaborazione soddisferebbe le caratteristiche del carico di lavoro in modo più preciso.

Vantaggi dell'adozione di questa best practice: identificando i requisiti di elaborazione e valutando le opzioni disponibili è possibile rendere il carico di lavoro più efficiente in termini di risorse.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per ottimizzare i carichi di lavoro cloud e ottenere prestazioni efficienti, è importante selezionare le opzioni di elaborazione più appropriate per il tuo caso d'uso e i requisiti di prestazioni. AWS offre una varietà di opzioni di elaborazione che soddisfano diversi carichi di lavoro nel cloud. Ad esempio, è possibile utilizzare Amazon EC2 per avviare e gestire server virtuali, AWS Lambda per eseguire codice senza dover effettuare il provisioning o gestire server, Amazon ECS o Amazon EKS per eseguire e gestire container oppure AWS Batch per elaborare grandi volumi di dati in parallelo. In base alle tue esigenze di dimensionamento ed elaborazione, scegli e configura la soluzione di elaborazione ottimale per la tua situazione. Puoi anche prendere in considerazione l'utilizzo di più tipi di soluzioni di elaborazione in un unico carico di lavoro in quanto ognuna ha i suoi vantaggi e svantaggi.

I passaggi seguenti ti guidano nella selezione delle opzioni di elaborazione giuste per soddisfare le caratteristiche del carico di lavoro e i requisiti prestazionali.

Passaggi dell'implementazione

- Comprendi i requisiti di elaborazione del tuo carico di lavoro. I requisiti essenziali da considerare includono le esigenze di elaborazione, gli schemi di traffico, gli schemi di accesso ai dati, le esigenze di dimensionamento e i requisiti di latenza.
- Scopri le diverse opzioni di elaborazione disponibili per il tuo carico di lavoro in AWS (come descritto in PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili. Ecco alcune importanti opzioni di elaborazione AWS, le caratteristiche e i casi d'uso più comuni:

AWS service	Key characteristics	Common use cases
Amazon Elastic Compute Cloud (Amazon EC2)	Has dedicated option for hardware, license requirements, large selection of different instance families, processor types and compute accelerators	Lift and shift migrations, monolithic application, hybrid environments, enterprise applications
Amazon Elastic Container Service (Amazon ECS),	Easy deployment, consistent environments, scalable	Microservices, hybrid environments

AWS service	Key characteristics	Common use cases
Amazon Elastic Kubernetes Service (Amazon EKS)		
AWS Lambda	Elaborazione serverless service that runs code in response to events and automatically manages the underlying compute resources.	Microservices, event-driven applications
AWS Batch	Efficiently and dynamically provisions and scales Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS), and AWS Fargate compute resources, with an option to use On-Demand or Spot Instances based on your job requirements	HPC, train ML models
Amazon Lightsail	Preconfigured Linux and Windows application for running small workloads	Simple web applications, custom website

- Valuta i costi (come la tariffa oraria o il trasferimento dei dati) e il sovraccarico di gestione (come l'applicazione di patch e il dimensionamento) associati a ciascuna opzione di elaborazione.
- Esegui esperimenti e benchmarking in un ambiente non di produzione per identificare quale opzione di elaborazione può soddisfare al meglio i requisiti del tuo carico di lavoro.
- Dopo aver sperimentato e identificato la tua nuova soluzione di calcolo, pianifica la migrazione e convalida i parametri prestazionali.
- Utilizza gli strumenti di monitoraggio AWS come Amazon CloudWatch e i servizi di ottimizzazione come AWS Compute Optimizer per ottimizzare continuamente le risorse di calcolo in base a modelli di utilizzo reali.

Risorse

Documenti correlati:

- Elaborazione nel cloud con AWS
- Tipi di istanza - Amazon EC2
- Amazon EKS Containers: Amazon EKS Worker Nodes
- Amazon ECS Containers: Amazon ECS Container Instances
- Configurazione della funzione Lambda
- Prescriptive Guidance for Containers (Guida prescrittiva per i container)
- Prescriptive Guidance for Serverless (Guida prescrittiva per serverless)

Video correlati:

- AWS re:Invent 2023 - AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 - New Amazon Elastic Compute Cloud generative AI capabilities in AMS
- AWS re:Invent 2023 - What's new with Amazon Elastic Compute Cloud
- AWS re:Invent 2023 - Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies
- AWS re:Invent 2021 - Powering next-gen Amazon Elastic Compute Cloud: Deep dive on the Nitro System
- AWS re:Invent 2019 - Optimize performance and cost for your AWS compute
- AWS re:Invent 2019 - Amazon Elastic Compute Cloud foundations
- AWS re:Invent 2022 - Deploy ML models for inference at high performance and low cost
- AWS re:Invent 2019 - Optimize performance and cost for your AWS compute
- Amazon EC2 foundations
- Deploy ML models for inference at high performance and low cost

Esempi correlati:

- Migrating the Web application to containers
- Esecuzione di un "Hello, World!" serverless
- Amazon EKS Workshop
- Amazon EC2 Workshop

- Efficient and Resilient Workloads with Amazon Elastic Compute Cloud Auto Scaling
- Migrating to AWS Graviton with Container Services

PERF02-BP02 Identificazione delle funzionalità e configurazione di calcolo disponibili

Comprendi le opzioni e le funzionalità di configurazione disponibili per il tuo servizio di elaborazione in modo da fornire la giusta quantità di risorse e migliorare l'efficienza delle prestazioni.

Anti-pattern comuni:

- Non valuti le opzioni di elaborazione o le famiglie di istanze disponibili rispetto alle caratteristiche del carico di lavoro.
- Esegui un provisioning eccessivo delle risorse di elaborazione per soddisfare i requisiti di picco della domanda.

Vantaggi dell'adozione di questa best practice: acquisisci familiarità con le funzionalità e le configurazioni di elaborazione di AWS in modo da poter utilizzare una soluzione di elaborazione ottimizzata per soddisfare le caratteristiche e le esigenze del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Ogni soluzione di elaborazione ha disponibili configurazioni e funzionalità specifiche per supportare caratteristiche e requisiti diversi del carico di lavoro. Scopri in che modo puoi completare al meglio il tuo carico di lavoro e quali opzioni di configurazione sono le migliori per la tua applicazione. Esempi di tali opzioni includono la famiglia di istanze, le dimensioni, le caratteristiche (GPU, I/O), il bursting, i timeout, le dimensioni delle funzioni, le istanze di container e la simultaneità. Se per il carico di lavoro è stata utilizzata la stessa opzione di calcolo per oltre quattro settimane e sai già che le caratteristiche resteranno uguali in futuro, puoi utilizzare AWS Compute Optimizer per scoprire se la tua attuale opzione di elaborazione è adatta ai carichi di lavoro dal punto di vista della CPU e della memoria.

Passaggi dell'implementazione

1. Comprendi i requisiti del carico di lavoro, come CPU, memoria e latenza.
2. Consulta la documentazione e le best practice AWS per scoprire le opzioni di configurazione consigliate che possono contribuire a migliorare le prestazioni dell'elaborazione. Ecco alcune opzioni di configurazione chiave da considerare:

Opzione di configurazione	Esempi
Tipo di istanza	• Le istanze ottimizzate per il calcolo sono l'ideale per i carichi di lavoro che richiedono un rapporto vCPU/memoria molto elevato. • Le istanze ottimizzate per la memoria offrono grandi quantità di memoria per carichi di lavoro intensivi in questo senso. • Le istanze ottimizzate per l'archiviazione sono progettate per carichi di lavoro che richiedono un accesso frequente e sequenziale in lettura e scrittura (IOPS) all'archiviazione locale.
Modello di prezzi	• Istanze on demand ti consentono di utilizzare la capacità di calcolo su base oraria o al secondo, senza impegni a lungo termine, e sono ideali per il bursting oltre le esigenze di base per le prestazioni. • Savings Plans offrono risparmi significativi rispetto alle istanze on demand in cambio dell'impegno a utilizzare una quantità specifica di potenza di elaborazione per un periodo di uno o tre anni. • istanze spot consentono di sfruttare la capacità inutilizzata delle istanze con uno sconto per i carichi di lavoro stateless e tolleranti ai guasti.
Auto Scaling	Utilizza Auto Scaling configurazione per abbinare le risorse di elaborazione ai modelli di traffico.

Opzione di configurazione	Esempi
Valutazione	• utilizza Compute Optimizer per ricevere un efficace suggerimento di machine learning riguardo alla configurazione più adatta alle tue caratteristiche di elaborazione. • utilizza AWS Lambda Power Tuning per selezionare la configurazione migliore per la tua funzione Lambda.
Acceleratori di calcolo basati su hardware	• Le istanze a calcolo accelerato eseguono funzioni come l'elaborazione grafica o la corrispondenza di schemi di dati in modo più efficiente rispetto alle alternative basate sulla CPU. • Per i carichi di lavoro di machine learning, sfrutta l'hardware specifico per il tuo carico di lavoro, come ad esempio AWS Trainium, AWS Inferentiae Amazon EC2 DL1

Risorse

Documenti correlati:

- Elaborazione in cloud con AWS
- Tipi di istanza Amazon EC2
- Controllo degli stati del processore dell'istanza Amazon EC2
- Amazon EKS Container: nodi worker di Amazon EKS
- Container Amazon ECS: Istanze di container di Amazon ECS
- Funzioni: configurazione della funzione Lambda

Video correlati:

- AWS re:Invent 2023 – AWS Graviton: The best price performance for your AWS workloads

- AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in AWS Management Console
- AWS re:Invent 2023 – What's new with Amazon EC2
- AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies
- AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System
- AWS re:Invent 2019 – Amazon EC2 foundations
- AWS re:Invent 2022 – https://www.youtube.com/watch?v=5B4-s_ivn1o

Esempi correlati:

- Codice dimostrativo di Compute Optimizer
- Workshop sulle istanze spot Amazon EC2
- Efficient and Resilient Workloads with Amazon EC2 AWS Auto Scaling
- Workshop per sviluppatori Graviton
- AWS for Microsoft workloads immersion day
- AWS for Linux workloads immersion day
- AWS Compute Optimizer Demo code (Codice dimostrativo di AWS Compute Optimizer)
- Workshop su Amazon EKS

PERF02-BP03 Raccolta dei parametri relativi al calcolo

Registra e monitora i parametri relativi all'elaborazione per comprendere meglio le prestazioni delle tue risorse di elaborazione e migliorarne le prestazioni e l'utilizzo.

Anti-pattern comuni:

- Utilizzi solo i file di log manuali per la ricerca dei parametri.
- Utilizzi solo i parametri predefiniti registrati dal software di monitoraggio.
- Rivedi i parametri solo quando c'è un problema.

Vantaggi dell'adozione di questa best practice: la raccolta dei parametri relativi alle prestazioni ti aiuta ad allineare le prestazioni delle applicazioni ai requisiti aziendali per garantire il rispetto delle esigenze dei carichi di lavoro. Può anche aiutarti a migliorare costantemente le prestazioni e l'utilizzo delle risorse del tuo carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

I carichi di lavoro del cloud possono generare grandi volumi di dati quali parametri, log ed eventi. Nel Cloud AWS, la raccolta dei parametri è un passaggio cruciale per migliorare la sicurezza, l'efficienza in termini di costi, le prestazioni e la sostenibilità. AWS fornisce un'ampia gamma di parametri relativi alle prestazioni utilizzando servizi di monitoraggio, come Amazon CloudWatch per fornirti approfondimenti preziosi. Parametri quali l'utilizzo della CPU, l'utilizzo della memoria, l'I/O del disco e il traffico di rete in entrata e in uscita possono fornire approfondimenti sui livelli di utilizzo o sui colli di bottiglia delle prestazioni. Utilizza tali parametri come parte di un approccio basato sui dati per ottimizzare e ottimizzare le risorse del tuo carico di lavoro. L'ideale sarebbe raccogliere tutti i parametri relativi alle tue risorse di elaborazione in un'unica piattaforma con policy di conservazione implementate per supportare costi e obiettivi operativi.

Passaggi dell'implementazione

1. Identifica quali parametri relativi alle prestazioni sono rilevanti per il tuo carico di lavoro. Raccogli i parametri sull'utilizzo delle risorse e sul modo in cui opera il tuo carico di lavoro nel cloud (come il tempo di risposta e la velocità di trasmissione effettiva).
   a. Parametri predefiniti di Amazon EC2
   b. Parametri predefiniti di Amazon ECS
   c. Parametri predefiniti di Amazon EKS
   d. Parametri predefiniti di Lambda
   e. Parametri di memoria e del disco di Amazon EC2
2. Scegli e configura la soluzione di registrazione e monitoraggio giusta per il tuo carico di lavoro.
   a. Osservabilità nativa di AWS
   b. AWS Distro for OpenTelemetry
   c. Amazon Managed Service for Prometheus
3. Definisci il filtro e l'aggregazione richiesti per i parametri in base ai requisiti del tuo carico di lavoro.
   a. Quantify custom application metrics with Amazon CloudWatch Logs and metric filters
   b. Collect custom metrics with Amazon CloudWatch strategic tagging
4. Configura le policy di conservazione dei dati per i parametri in modo che corrispondano ai tuoi obiettivi operativi e di sicurezza.
   a. Conservazione dei dati predefinita per i parametri CloudWatch

   b. Conservazione dei dati predefinita per CloudWatch Logs
5. Se necessario, crea allarmi e notifiche per i parametri in modo da rispondere in modo proattivo ai problemi relativi alle prestazioni.
   a. Create alarms for custom metrics using Amazon CloudWatch anomaly detection
   b. Create metrics and alarms for specific web pages with Amazon CloudWatch RUM
6. Usa l'automazione per implementare gli agenti di aggregazione di parametri e log.
   a. Automazione AWS Systems Manager
   b. OpenTelemetry Collector

Risorse

Documenti correlati:

- Monitoraggio e osservabilità
- Best practices: implementing observability with AWS
- Documentazione di Amazon CloudWatch
- Raccolta di parametri e registri da istanze Amazon EC2 e da server on-premise con l'agente di CloudWatch
- Accesso a Amazon CloudWatch Logs per AWS Lambda
- Utilizzo di CloudWatch Logs con istanze di container
- Pubblicazione di parametri personalizzati
- AWS Answers: Centralized Logging (AWS Answers: registrazione centralizzata)
- Servizi AWS che pubblicano parametri CloudWatch
- Monitoraggio di Amazon EKS su AWS Fargate

Video correlati:

- AWS re:Invent 2023 – [LAUNCH] Application monitoring for modern workloads
- AWS re:Invent 2023 – Implementing application observability
- AWS re:Invent 2023 – Building an effective observability strategy
- AWS re:Invent 2023 – Seamless observability with AWS Distro for OpenTelemetry
- Application Performance Management on AWS

Esempi correlati:

- AWS for Linux Workloads Immersion Day- Amazon CloudWatch
- Monitoring Amazon ECS clusters and containers
- Monitoring with Amazon CloudWatch dashboards
- Workshop su Amazon EKS

PERF02-BP04 Configurazione e dimensionamento corretto delle risorse di elaborazione

Configura e dimensiona correttamente le risorse di elaborazione per soddisfare i requisiti di prestazioni del carico di lavoro ed evitare un utilizzo insufficiente o eccessivo delle risorse.

Anti-pattern comuni:

- Ignori i requisiti di prestazioni del carico di lavoro, con il risultato del provisioning eccessivo o insufficiente delle risorse di elaborazione.
- Scegli semplicemente l'istanza più grande o più piccola disponibile per tutti i carichi di lavoro.
- Usi una sola famiglia di istanze per semplificare la gestione.
- Ignori i suggerimenti di AWS Cost Explorer o Compute Optimizer per il corretto dimensionamento.
- Non rivaluti il carico di lavoro in base all'idoneità dei nuovi tipi di istanza.
- Certifichi solo un numero limitato di configurazioni di istanza per l'organizzazione.

Vantaggi dell'adozione di questa best practice: il corretto dimensionamento delle risorse di elaborazione garantisce un funzionamento ottimale nel cloud evitando il provisioning eccessivo o insufficiente delle risorse. Il corretto dimensionamento delle risorse di elaborazione comporta in genere prestazioni ottimali e una migliore esperienza cliente, riducendo al contempo i costi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Il dimensionamento corretto consente alle organizzazioni di gestire la propria infrastruttura cloud in modo efficiente ed economico, rispettando al contempo le esigenze aziendali. Un provisioning eccessivo delle risorse cloud può comportare costi aggiuntivi, mentre un provisioning insufficiente può comportare prestazioni scadenti e un'esperienza negativa per il cliente. AWS fornisce strumenti come AWS Compute Optimizer e AWS Trusted Advisor che utilizzano dati storici per fornire consigli per dimensionare correttamente le risorse di elaborazione.

Passaggi dell'implementazione

- Scegli il tipo di istanza più adatto alle tue esigenze:
  - Come faccio a scegliere il tipo di istanza Amazon EC2 appropriato per il mio carico di lavoro?
  - Selezione del tipo di istanza basata sugli attributi per il parco istanze Amazon EC2
  - Create an Auto Scaling group using attribute-based instance type selection
  - Optimizing your Kubernetes compute costs with Karpenter consolidation
- Analizza le varie caratteristiche di prestazione del tuo carico di lavoro e come queste sono correlate a memoria, rete e utilizzo della CPU. Utilizza questi dati per scegliere le risorse che meglio corrispondono al profilo del tuo carico di lavoro e agli obiettivi di prestazioni.
- Monitora l'utilizzo delle risorse con gli strumenti di monitoraggio di AWS come Amazon CloudWatch.
- Seleziona la configurazione corretta per la risorsa di elaborazione.
  - Per i carichi di lavoro effimeri, valuta le metriche Amazon CloudWatch dell'istanza , ad esempio `CPUUtilization` per identificare se l'istanza è sottoutilizzata o sovrautilizzata.
  - Per i carichi di lavoro stabili, esegui i controlli con gli strumenti di ridimensionamento di AWS, come AWS Compute Optimizer e AWS Trusted Advisor a intervalli regolari per individuare le opportunità di ottimizzazione e ridimensionamento della risorsa di elaborazione.
- Esegui il test delle modifiche apportate alla configurazione in un ambiente non di produzione prima di implementarle in un ambiente live.
- Rivaluta costantemente nuove offerte di elaborazione e confrontale con le esigenze del carico di lavoro.

Risorse

Documenti correlati:

- Elaborazione in cloud con AWS
- Tipi di istanza Amazon EC2
- Container Amazon ECS: Istanze di container di Amazon ECS
- Amazon EKS Container: nodi worker di Amazon EKS
- Funzioni: configurazione della funzione Lambda
- Controllo degli stati del processore dell'istanza Amazon EC2

Video correlati:

- Amazon EC2 foundations
- AWS re:Invent 2023 – AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in AWS Management Console
- AWS re:Invent 2023 – What's new with Amazon EC2
- AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies
- AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System
- AWS re:Invent 2019 – Amazon EC2 foundations

Esempi correlati:

- AWS Compute Optimizer Demo code (Codice dimostrativo di AWS Compute Optimizer)
- Workshop su Amazon EKS
- Right-sizing recommendations

PERF02-BP05 Dimensionamento dinamico delle risorse di elaborazione

Sfrutta l'elasticità del cloud per dimensionare dinamicamente le risorse di elaborazione per soddisfare le tue esigenze ed evitare un provisioning eccessivo o insufficiente per il tuo carico di lavoro.

Anti-pattern comuni:

- Risposta agli allarmi aumentando manualmente la capacità.
- Utilizzi le stesse linee guida per il dimensionamento (generalmente infrastruttura statica) di quelle on-premise.
- Mantenimento della maggiore capacità dopo un evento di dimensionamento, senza ripristinare quella originale.

Vantaggi dell'adozione di questa best practice: La configurazione e il test dell'elasticità delle risorse di elaborazione possono aiutarti a risparmiare denaro, mantenere i benchmark delle prestazioni e migliorare l'affidabilità al variare del traffico.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

AWS offre la flessibilità necessaria per dimensionare le risorse in modo dinamico attraverso una varietà di meccanismi di dimensionamento per soddisfare le variazioni della domanda. In combinazione con i parametri relativi all'elaborazione, il dimensionamento dinamico consente ai carichi di lavoro di rispondere automaticamente alle modifiche e utilizzare il set ottimale di risorse di elaborazione per raggiungere l'obiettivo.

Puoi adottare varie strategie di approccio per associare l'offerta di risorse alla domanda.

- Approccio al tracciamento degli obiettivi: monitora il parametro di dimensionamento e aumenta o diminuisci automaticamente la capacità in base alle esigenze.
- Dimensionamento predittivo: dimensiona in previsione delle tendenze giornaliere e settimanali.
- Approccio basato sulla pianificazione: imposta il tuo programma di dimensionamento in base alle variazioni di carico prevedibili.
- Scalabilità del servizio: scegli i servizi (come quelli serverless) che si dimensionano automaticamente per progettazione.

Assicurati che le distribuzioni dei carichi di lavoro siano in grado di gestire eventi di dimensionamento.

Passaggi dell'implementazione

- Istanze di elaborazione, container e funzioni forniscono tutti meccanismi di elasticità, in combinazione con il dimensionamento automatico o sotto forma di funzionalità del servizio. Ecco alcuni esempi di meccanismi di dimensionamento automatico:

Meccanismo di scalabilità automatica	Dove usare
Amazon EC2 Auto Scaling	Per assicurarti di avere il numero corretto di Amazon EC2 istanze disponibili per gestire il carico utente per la tua applicazione.
Application Auto Scaling	per dimensionare automaticamente le risorse per servizi AWS diversi da Amazon EC2, ad esempio AWS Lambda funzioni o Amazon Elastic Container Service (Amazon ECS) servizi.

Meccanismo di scalabilità automatica	Dove usare
Kubernetes Cluster Autoscaler/Karpenter	Per dimensionare automaticamente i cluster Kubernetes.

- Si parla spesso di dimensionamento con servizi di elaborazione come le istanze Amazon EC2 o le funzioni AWS Lambda. Assicurati di considerare anche la configurazione di servizi non di elaborazione come AWS Glue per soddisfare la domanda.
- Verifica che i parametri per il dimensionamento corrispondano alle caratteristiche del carico di lavoro da implementare. Se distribuisci un'applicazione di transcodifica video, è previsto il 100% di utilizzo della CPU e non deve essere il parametro principale. Utilizza la profondità della coda dei processi di transcodifica. Puoi utilizzare una metrica personalizzata per la tua politica di scalabilità, se necessario. Per scegliere la metrica corretta, consulta le linee guida seguenti per Amazon EC2:
  - La metrica deve essere una metrica di utilizzo valida e descrivere il livello di impiego di un'istanza.
  - Il valore della metrica deve aumentare o diminuire proporzionalmente in base al numero di istanze nel gruppo con Auto Scaling.
- Assicurati di utilizzare il dimensionamento dinamico invece del dimensionamento manuale per il gruppo con Auto Scaling in uso. È consigliabile utilizzare le policy di dimensionamento del monitoraggio degli obiettivi nel dimensionamento dinamico.
- Verifica che le implementazioni dei carichi di lavoro siano in grado di gestire entrambi gli eventi di dimensionamento (aumento e riduzione). Ad esempio, puoi usare la cronologia delle attività per verificare un'attività di dimensionamento per un gruppo Auto Scaling.
- Analizza il tuo carico di lavoro per individuare modelli prevedibili e dimensionare le tue risorse in modo proattivo, anticipando variazioni nella domanda previste e pianificate. Con il dimensionamento predittivo puoi eliminare la necessità di offrire capacità in eccedenza. Per ulteriori informazioni, consulta Dimensionamento predittivo con Amazon EC2 Auto Scaling.

Risorse

Documenti correlati:

- Elaborazione in cloud con AWS
- Tipi di istanza Amazon EC2
- Container Amazon ECS: Istanze di container di Amazon ECS
- Amazon EKS Container: nodi worker di Amazon EKS

- Funzioni: configurazione della funzione Lambda
- Controllo degli stati del processore dell'istanza Amazon EC2
- Deep Dive on Amazon ECS Cluster Auto Scaling
- Introducing Karpenter – An Open-Source High-Performance Kubernetes Cluster Autoscaler

Video correlati:

- AWS re:Invent 2023 – AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in AWS Management Console
- AWS re:Invent 2023 – What's new with Amazon EC2
- AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies
- AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System
- AWS re:Invent 2019 – Amazon EC2 foundations

Esempi correlati:

- Esempi di gruppo di Amazon EC2 Auto Scaling
- Workshop su Amazon EKS
- Scale your Amazon EKS workloads by running on IPv6

PERF02-BP06 Uso di acceleratori di elaborazione ottimizzati basati su hardware

Usa gli acceleratori hardware per eseguire determinate funzioni in modo più efficiente rispetto alle alternative basate sulla CPU.

Anti-pattern comuni:

- Nel carico di lavoro non hai confrontato un'istanza generica con un'istanza dedicata in grado di offrire prestazioni più elevate e costi inferiori.
- Usi gli acceleratori di calcolo basati su hardware per attività in cui sono più efficienti le alternative basate su CPU.
- Utilizzo delle GPU non monitorato.

Vantaggi dell'adozione di questa best practice: utilizzando gli acceleratori basati su hardware, come le unità di elaborazione grafica (GPU) e gli FPGA (Field Programmable Gate Array), è possibile eseguire determinate funzioni di elaborazione in modo più efficiente.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Le istanze a calcolo accelerato forniscono l'accesso agli acceleratori di calcolo basati su hardware, come GPU e FPGA. Questi acceleratori hardware eseguono alcune funzioni, come l'elaborazione grafica o la rilevazione della corrispondenza dei modelli di dati, in modo più efficiente rispetto alle alternative basate su CPU. Molti carichi di lavoro accelerati, come il rendering grafico, la transcodifica e il machine learning, sono altamente variabili in termini di utilizzo di risorse. Esegui questo hardware solo per il tempo necessario e disattivalo con l'automazione quando non serve per migliorare l'efficienza complessiva delle prestazioni.

Passaggi dell'implementazione

- Identifica quali istanze a calcolo accelerato possono soddisfare i tuoi requisiti.
- Per i carichi di lavoro di machine learning, sfrutta l'hardware specifico per le tue esigenze, ad esempio AWS Trainium, AWS Inferentia e Amazon EC2 DL1. Le istanze AWS Inferentia come Inf2 offrono fino al 50% in più di prestazioni/watt rispetto alle istanze Amazon EC2 paragonabili.
- Raccogli i parametri di utilizzo delle istanze a calcolo accelerato. Ad esempio, puoi utilizzare l'agente CloudWatch per raccogliere parametri come `utilization_gpu` e `utilization_memory` per le GPU come mostrato in Raccolta dei parametri delle GPU NVIDIA con Amazon CloudWatch.
- Ottimizza il codice, il funzionamento della rete e le impostazioni degli acceleratori hardware per garantire il pieno utilizzo dell'hardware sottostante.
  - Ottimizza l impostazioni delle GPU
  - Monitoraggio e ottimizzazione delle GPU nell'AMI per il Deep Learning
  - Ottimizzazione dell'I/O per la messa a punto delle prestazioni delle GPU dedicate all'addestramento del deep learning in Amazon SageMaker
- Utilizzate le librerie e i driver per GPU più recenti e performanti.
- Utilizza l'automazione per rilasciare le istanze GPU non in uso.

Risorse

Documenti correlati:

- Working with GPUs on Amazon Elastic Container Service
- Istanze GPU
- Istanze con AWS Trainium
- Istanze con AWS Inferentia
- Let's Architect! Architecting with custom chips and accelerators

- Calcolo accelerato
- Istanze Amazon EC2 VT1
- How do I choose the appropriate Amazon EC2 instance type for my workload?
- Choose the best AI accelerator and model compilation for computer vision inference with Amazon SageMaker

Video correlati:

- AWS re:Invent 2021 - How to select Amazon Elastic Compute Cloud GPU instances for deep learning
- AWS re:Invent 2022 - [NEW LAUNCH!] Introducing AWS Inferentia2-based Amazon EC2 Inf2 instances
- AWS re:Invent 2022 - Accelerate deep learning and innovate faster with AWS Trainium
- AWS re:Invent 2022 - Deep learning on AWS with NVIDIA: From training to deployment

Esempi correlati:

- Amazon SageMaker and NVIDIA GPU Cloud (NGC)
- Use SageMaker with Trainium and Inferentia for optimized deep learning training and inferencing workloads
- Optimizing NLP models with Amazon Elastic Compute Cloud Inf1 instances in Amazon SageMaker

Gestione dati

PERF 3. In che modo archivi, gestisci e accedi ai dati nel tuo carico di lavoro?

La soluzione ottimale per la gestione dei dati in un sistema specifico varia in base al tipo di dati (blocco, file o oggetto), agli schemi di accesso (casuali o sequenziali), alla velocità di trasmissione effettiva necessaria, alla frequenza di accesso (online, offline, archivio), alla frequenza di aggiornamento (WORM, dinamico) e ai vincoli di disponibilità e durata. I carichi di lavoro Well-Architected utilizzano archivi dati appositamente progettati che impiegano diverse funzionalità per migliorare le prestazioni.

Best practice

- PERF03-BP01 Uso di un archivio dati dedicato che supporta al meglio i requisiti di accesso e archiviazione dei dati
- PERF03-BP02 Valutazione delle opzioni di configurazione disponibili per datastore
- PERF03-BP03 Raccolta e registrazione dei parametri delle prestazioni del datastore
- PERF03-BP04 Implementazione di strategie per migliorare le prestazioni delle query nel datastore
- PERF03-BP05 Implementa modelli di accesso ai dati che utilizzano la memorizzazione nella cache

PERF03-BP01 Uso di un archivio dati dedicato che supporta al meglio i requisiti di accesso e archiviazione dei dati

Comprendi le caratteristiche dei dati (come la condivisione, le dimensioni, la dimensione della cache, gli schemi di accesso, la latenza, la velocità di trasmissione effettiva e la persistenza dei dati) per selezionare i data store (archiviazione o database) dedicati per il tuo carico di lavoro.

Anti-pattern comuni:

- Continui a utilizzare un datastore per via dell'esperienza e delle competenze interne relative a quel particolare tipo di soluzione di database.
- Ritieni che tutti i carichi di lavoro abbiano requisiti di accesso e archiviazione dei dati simili.
- Non hai implementato un catalogo di dati per eseguire l'inventario dei tuoi asset.

Vantaggi dell'adozione di questa best practice: la comprensione delle caratteristiche e dei requisiti dei dati ti consente di determinare la tecnologia di archiviazione più efficiente e performante appropriata per le esigenze del tuo carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Quando selezioni e implementi l'archiviazione di dati, assicurati che le caratteristiche di query, dimensionamento e archiviazione supportino i requisiti dei dati del carico di lavoro. AWS fornisce numerose tecnologie di database e archiviazione di dati, tra cui archiviazione a blocchi, archiviazione di oggetti, archiviazione di streaming, file system, database relazionali, chiave-valore, di documenti, in memoria, a grafo, di serie temporali e di libro mastro. Ogni soluzione di gestione dei dati offre soluzioni e configurazioni adatte a gestire i tuoi casi d'uso e modelli di dati. Comprendendo le caratteristiche e i requisiti dei dati, puoi abbandonare la tecnologia di archiviazione monolitica e gli approcci restrittivi e validi per tutti, per concentrarti sulla gestione dei dati in modo appropriato.

Passaggi dell'implementazione

- Esegui un inventario dei vari tipi di dati esistenti nel tuo carico di lavoro.
- Comprendi e documenta le caratteristiche e i requisiti dei dati, tra cui:
  - Tipo di dati (non strutturati, semi-strutturati, relazionali)
  - Volume e crescita dei dati
  - Durabilità dei dati: persistenti, effimeri, transitori
  - Requisiti ACID (atomicità, coerenza, isolamento, durabilità)
  - Schemi di accesso ai dati (con uso intensivo di lettura o scrittura)
  - Latenza
  - Throughput
  - IOPS (operazioni di input/output al secondo)
  - Periodo di conservazione dei dati
- Scopri i diversi archivi di dati (servizi di database e archiviazione) disponibili per il carico di lavoro AWS che possono soddisfare le caratteristiche dei tuoi dati, come descritto in PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili. Alcuni esempi di tecnologie di archiviazione AWS e delle loro caratteristiche chiave sono:

Tipo	Servizi AWS	Caratteristiche chiave
Object storage	Amazon S3	Unlimited scalability, high availability, and multiple options for accessibility.

Tipo	Servizi AWS	Caratteristiche chiave
		Transferring and accessing objects in and out of Amazon S3 can use a service, such as Accelerazione del trasferimento or Punti di accesso, to support your location, security needs, and access patterns.
Archiving storage	Amazon S3 Glacier	Built for data archiving.
Streaming storage	Amazon Kinesis Amazon Managed Streaming for Apache Kafka (Amazon MSK)	Efficient ingestion and storage of streaming data.
Shared file system	Amazon Elastic File System (Amazon EFS)	File system montabile a cui è possibile accedere da più tipi di soluzioni di calcolo.
Shared file system	Amazon FSx	Built on the latest AWS compute solutions to support four commonly used file systems: NetApp ONTAP, OpenZFS, Windows File Server, and Lustre. Amazon FSx , la latenza, la velocità di trasmissione effettiva e le operazioni di input/output al secondo (IOPS) vary per file system and should be considered when selecting the right file system for your workload needs.

Tipo	Servizi AWS	Caratteristiche chiave
Block storage	Amazon Elastic Block Store (Amazon EBS)	Scalable, high-performance block-storage service designed for Amazon Elastic Compute Cloud (Amazon EC2). Amazon EBS includes SSD-backed storage for transactional, IOPS-intensive workloads and HDD-backed storage for throughput-intensive workloads.
Relational database	Amazon Aurora, Amazon RDS, Amazon Redshift.	Designed to support ACID (atomicity, consistency, isolation, durability) transactions, and maintain referential integrity and strong data consistency. Many traditional applications, enterprise resource planning (ERP), customer relationship management (CRM), and ecommerce use relational databases to store their data.
Key-value database	Amazon DynamoDB	Optimized for common access patterns, typically to store and retrieve large volumes of data. High-traffic web apps, ecommerce systems, and gaming applications are typical use-cases for key-value databases.

Tipo	Servizi AWS	Caratteristiche chiave
Document database	Amazon DocumentDB	Designed to store semi-stru ctured data as JSON-like documents. These databases help developers build and update applications such as content management, catalogs, and user profiles quickly.
In-memory database	Amazon ElastiCache , Amazon MemoryDB per Redis	Used for applications that require real-time access to data, lowest latency and highest throughput. You may use in-memory databases for application caching, session management, gaming leaderboards, low latency ML feature store, microserv ices messaging system, and a high-throughput streaming mechanism
Graph database	Amazon Neptune	Used for applications that must navigate and query millions of relationships between highly connected graph datasets with milliseco nd latency at large scale. Many companies use graph databases for fraud detection , social networking, and recommendation engines.

Tipo	Servizi AWS	Caratteristiche chiave
Time Series database	Amazon Timestream	Used to efficiently collect, synthesize, and derive insights from data that changes over time. IoT applications, DevOps, and industrial telemetry can utilize time-series databases.
Wide column	Amazon Keyspaces (per Apache Cassandra)	Uses tables, rows, and columns, but unlike a relational database, the names and format of the columns can vary from row to row in the same table. You typically see a wide column store in high scale industrial apps for equipment maintenance, fleet management, and route optimization.
Ledger	Amazon Quantum Ledger Database (Amazon QLDB)	Provides a centralized and trusted authority to maintain a scalable, immutable, and cryptographically verifiabl e record of transactions for every application. We see ledger databases used for systems of record, supply chain, registrations, and even banking transactions.

- Se stai creando una piattaforma dati, sfrutta la moderna architettura dei dati AWS per integrare data lake, data warehouse e archivi di dati dedicati.
- Le domande chiave da porsi quando si sceglie un data store per il carico di lavoro sono le seguenti:

Question	Things to consider
How is the data structured?	• Se i dati non sono strutturati, prendi in considerazione un archivio di oggetti come Amazon S3 o un database NoSQL come Amazon DocumentDB • Per i dati chiave-valore, valuta DynamoDB, Amazon ElastiCache for Redis o Amazon MemoryDB for Redis
What level of referential integrity is required?	• Per i vincoli di chiave esterna, i database relazionali come Amazon RDS e Aurora possono fornire questo livello di integrità. • In genere, in un modello di dati NoSQL, i dati vengono denormalizzati in un singolo documento o in una raccolta di documenti da recuperare in un'unica richiesta, anziché essere uniti tra diversi documenti o tabelle.
Is ACID (atomicity, consistency, isolation, durability) compliance required?	• Se sono necessarie proprietà ACID associate ai database relazionali, valuta un database relazionale come Amazon RDS e Aurora. • Se è necessaria un'elevata consistenza per il database NoSQL, puoi utilizzare l'elevata consistenza di lettura di DynamoDB.

Question	Things to consider
How will the storage requirements change over time? How does this impact scalability?	• Database serverless come DynamoDB e Amazon Quantum Ledger Database (Amazon QLDB) possono dimensionarsi dinamicamente. • Per i database relazionali sono previsti limiti massimi per l'archiviazione assegnata, al raggiungimento dei quali si rende spesso necessario partizionare orizzontalmente tali database tramite meccanismi quali lo sharding.
What is the proportion of read queries in relation to write queries? Would caching be likely to improve performance?	• I carichi di lavoro con molte operazioni di lettura possono trarre vantaggio da un livello di caching, ad esempio ElastiCache o DAX se il database è DynamoDB. • È anche possibile passare le operazioni di lettura alle repliche di lettura con database relazionali come Amazon RDS.
Does storage and modification (OLTP - Online Transaction Processing) or retrieval and reporting (OLAP - Online Analytical Processing) have a higher priority?	• Per un'elaborazione transazionale letta così com'è ad alta velocità di trasmissione effettiva, prendi in considerazione un database NoSQL come DynamoDB. • Per schemi di lettura complessi con velocità di trasmissione effettiva elevata (come il join) con un uso coerente di Amazon RDS. • Per le query analitiche, prendi in considerazione un database colonnare come Amazon Redshift o l'esportazione dei dati in Amazon S3 e l'esecuzione di analisi utilizzando Athena o Amazon QuickSight.

Question	Things to consider
What level of durability does the data require?	• Aurora replica automaticamente i dati su tre zone di disponibilità all'interno di una Regione, il che significa che i dati sono altamente durevoli con minori probabilità di perdite. • DynamoDB viene automaticamente replicato in più zone di disponibilità per offrire livelli elevati di disponibilità e durabilità dei dati. • Amazon S3 offre il 99,999999999 di durabilità. Molti servizi di database, come Amazon RDS e DynamoDB, supportano l'esportazione di dati su Amazon S3 per la conservazione e l'archiviazione a lungo termine.
Is there a desire to move away from commercial database engines or licensing costs?	• Valuta motori open-source come PostgreSQL e MySQL su Amazon RDS o Aurora. • Usa AWS Database Migration Service e AWS Schema Conversion Tool per eseguire le migrazioni dai motori di database commerciali a quelli open-source.
What is the operational expectation for the database? Is moving to managed services a primary concern?	• Utilizzare Amazon RDS, invece di Amazon EC2, e scegliere DynamoDB o Amazon DocumentDB anziché ospitare in autonomia un database NoSQL, riduce le spese operative.

Question	Things to consider
How is the database currently accessed? Is it only application access, or are there business intelligence (BI) users and other connected off-the-shelf applications?	• Se fossero presenti dipendenze verso altri strumenti esterni, potresti dover mantenere la compatibilità con i database che essi supportano. Amazon RDS è completam ente compatibile con le diverse versioni dei motori che supporta, compresi Microsoft SQL Server, Oracle, MySQL e PostgreSQL.

- Esegui esperimenti e benchmarking in un ambiente non di produzione per identificare quale datastore può soddisfare al meglio i requisiti del tuo carico di lavoro.

Risorse

Documenti correlati:

- Tipi di volume Amazon EBS
- Archiviazione Amazon EC2
- Amazon EFS: Amazon EFS Performance
- Amazon FSx for Lustre Performance
- Amazon FSx for Windows File Server Performance
- DocumentazioneAmazon S3 Glacier: S3 Glacier
- Amazon S3: Request Rate and Performance Considerations
- Archiviazione nel cloud in AWS
- Caratteristiche e monitoraggio degli I/O - Amazon EBS
- Database su AWS Cloud
- AWS Database Caching
- DynamoDB Accelerator
- Best practice con Amazon Aurora
- Prestazioni Amazon Redshift
- Amazon Athena top 10 performance tips
- Amazon Redshift Spectrum best practices
- Amazon DynamoDB best practices

- Choose between Amazon EC2 and Amazon RDS
- Best practice e strategie di caching - Amazon ElastiCache

Video correlati:

- AWS re:Invent 2023: Improve Amazon Elastic Block Store efficiency and be more cost-efficient
- AWS re:Invent 2023: Optimizing storage price and performance with Amazon Simple Storage Service
- AWS re:Invent 2023: Building and optimizing a data lake on Amazon Simple Storage Service
- AWS re:Invent 2022: Building modern data architectures on AWS
- AWS re:Invent 2022: Building data mesh architectures on AWS
- AWS re:Invent 2023: Deep dive into Amazon Aurora and its innovations
- AWS re:Invent 2023: Advanced data modeling with Amazon DynamoDB
- AWS re:Invent 2022: Modernize apps with purpose-built databases
- Amazon DynamoDB deep dive: Advanced design patterns

Esempi correlati:

- AWS Purpose Built Databases Workshop
- Databases for Developers
- AWS Modern Data Architecture Immersion Day
- Build a Data Mesh on AWS
- Esempi di Amazon S3
- Optimize Data Pattern using Amazon Redshift Data Sharing
- Migrazioni dei database
- MS SQL Server - AWS Database Migration Service (AWS DMS) Replication Demo
- Database Modernization Hands On Workshop (Workshop pratico sulla modernizzazione dei database)
- Amazon Neptune Esempi

PERF03-BP02 Valutazione delle opzioni di configurazione disponibili per datastore

Comprendi e valuta le varie funzionalità e opzioni di configurazione disponibili per i tuoi datastore per ottimizzare lo spazio di archiviazione e le prestazioni per il tuo carico di lavoro.

Anti-pattern comuni:

- Utilizzi un solo tipo di storage, ad esempio Amazon EBS, per tutti i carichi di lavoro.
- Utilizzi la capacità di IOPS allocata per tutti i carichi di lavoro senza test reali su tutti i livelli di archiviazione.
- Non conosci le opzioni di configurazione della soluzione di gestione dei dati scelta.
- Ti basi soltanto sull'aumento delle dimensioni dell'istanza, senza tenere conto di altre opzioni di configurazione disponibili.
- Non esegui il test delle caratteristiche di dimensionamento del tuo datastore.

Vantaggi dell'adozione di questa best practice: l'esplorazione e la sperimentazione delle configurazioni dei datastore ti consentono di ridurre il costo dell'infrastruttura, migliorare le prestazioni e diminuire le attività richieste per mantenere i carichi di lavoro.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Un carico di lavoro può utilizzare uno o più datastore in base ai requisiti di archiviazione e accesso ai dati. Per ottimizzare prestazioni, efficienza e costi, è necessario valutare gli schemi di accesso ai dati per determinare le configurazioni appropriate del datastore. Nella valutazione delle opzioni di datastore, prendi in considerazione vari aspetti come le opzioni di archiviazione, la memoria, l'elaborazione, la replica di lettura, i requisiti di coerenza, il pool di connessioni e le opzioni di caching. Esegui esperimenti con queste diverse opzioni di configurazione per migliorare i parametri di efficienza delle prestazioni.

Passaggi dell'implementazione

- Esamina le configurazioni correnti (come il tipo di istanza, la dimensione di archiviazione o la versione del motore di database) del tuo datastore.
- Consulta la documentazione e le best practice AWS per scoprire le opzioni di configurazione consigliate che possono contribuire a migliorare le prestazioni del datastore. Le principali opzioni da considerare per il datastore sono le seguenti:

Configuration option	Examples
Offloading reads (like read replicas and caching)	• Per le tabelle DynamoDB, è possibile rimuovere le operazioni di lettura grazie a DAX per il caching. • Puoi creare un cluster Amazon ElastiCac he for Redis e configurare l'applicazione in modo che legga prima dalla cache e quindi passi al database se l'elemento richiesto non è presente. • I database relazionali come Amazon RDS e Aurora, nonché i database NoSQL allocati, come Neptune e Amazon DocumentDB, supportano tutti l'aggiunta di repliche di lettura per rimuovere le operazioni di lettura del carico di lavoro. • I database serverless come DynamoDB si dimensionano automaticamente. Assicurat i di avere abbastanza unità di capacità di lettura (RCU) assegnate per gestire il carico di lavoro.

Configuration option	Examples
Scaling writes (like partition key sharding or introducing a queue)	• Per i database relazionali, è possibile aumentare la dimensione dell'istanza per gestire un maggiore carico di lavoro o aumentare la capacità di IOPS allocata per gestire una maggiore velocità di trasmissione effettiva verso l'archiviazione sottostante. • È anche possibile introdurre una coda davanti al database, invece di eseguire direttamente la scrittura su di esso. Questo schema consente di disaccoppiare l'acquisizione dal database e controllare il flusso, in modo che il database sia in grado di gestirlo. • Raggruppare in batch le richieste di scrittura, anziché creare molte transazioni di breve durata, può aiutare a migliorare la velocità di trasmissione effettiva in database relazionali con un elevato volume in scrittura. • I database serverless come DynamoDB possono dimensionare automaticamente la velocità di trasmissione effettiva in scrittura oppure è possibile regolare le unità di capacità in scrittura (WCU) assegnate, a seconda della modalità di capacità. • È tuttavia possibile che si verifichino problemi con le partizioni hot quando si raggiungono i limiti di velocità di trasmissione effettiva per una determinata chiave di partizione. Questo problema può essere arginato scegliendo una chiave di partizione con una distribuzione più uniforme o

Configuration option	Examples
	eseguendo lo sharding in lettura della chiave di partizione.
Policies to manage the lifecycle of your datasets	• È possibile utilizzare Amazon S3 Lifecycle per gestire gli oggetti durante il loro ciclo di vita. Se gli schemi di accesso sono sconosciuti, mutevoli o imprevedibili, puoi usare Amazon S3 Intelligent-Tiering, che monitora gli schemi di accesso e sposta automaticamente gli oggetti che non hanno fatto registrare accessi a costi contenuti . Puoi sfruttare i parametri di Amazon S3 Storage Lens per identificare le opportunità di ottimizzazione e le lacune nella gestione del ciclo di vita. • La gestione del ciclo di vita di Amazon EFS gestisce automaticamente l'archiviazione dei file per i tuoi file system.
Connection management and pooling	• È possibile utilizzare Server proxy per Amazon RDS con Amazon RDS e Aurora per gestire le connessioni al database. • I database serverless come DynamoDB non hanno connessioni associate, ma valuta la capacità assegnata e le policy di dimension amento automatico per affrontare i picchi nel carico.

- Esegui esperimenti e benchmarking in un ambiente non di produzione per identificare quale opzione di configurazione può soddisfare i requisiti del tuo carico di lavoro.
- Dopo aver sperimentato, pianifica la migrazione e convalida i parametri delle prestazioni.
- Usa gli strumenti AWS per il monitoraggio, come Amazon CloudWatch, e l'ottimizzazione, come Amazon S3 Storage Lens, per ottimizzare continuamente il tuo datastore utilizzando schemi di utilizzo reali.

Risorse

Documenti correlati:

- Archiviazione nel cloud in AWS
- Tipi di volume Amazon EBS
- Archiviazione Amazon EC2
- Amazon EFS: Amazon EFS Performance
- Amazon FSx for Lustre Performance
- Amazon FSx for Windows File Server Performance
- DocumentazioneAmazon S3 Glacier: S3 Glacier
- Amazon S3: Request Rate and Performance Considerations
- Caratteristiche e monitoraggio degli I/O - Amazon EBS
- Database su AWS Cloud
- AWS Database Caching
- DynamoDB Accelerator
- Best practice con Amazon Aurora
- Prestazioni Amazon Redshift
- Amazon Athena top 10 performance tips
- Amazon Redshift Spectrum best practices
- Amazon DynamoDB best practices

Video correlati:

- AWS re:Invent 2023: Improve Amazon Elastic Block Store efficiency and be more cost-efficient
- AWS re:Invent 2023: Optimize storage price and performance with Amazon Simple Storage Service
- AWS re:Invent 2023: Building and optimizing a data lake on Amazon Simple Storage Service
- AWS re:Invent 2023: What's new with AWS file storage
- AWS re:Invent 2023: Dive deep into Amazon DynamoDB

Esempi correlati:

- AWS Purpose Built Databases Workshop
- Databases for Developers
- AWS Modern Data Architecture Immersion Day
- Amazon EBS Autoscale
- Esempi di Amazon S3
- Amazon DynamoDB Examples
- AWS Database migration samples
- Database Modernization Workshop (Workshop sulla modernizzazione dei database)
- Working with parameters on your Amazon RDS for Postgress DB

PERF03-BP03 Raccolta e registrazione dei parametri delle prestazioni del datastore

Tieni traccia e registra i parametri delle prestazioni pertinenti per il tuo datastore per capire l'andamento delle prestazioni delle soluzioni di gestione dei dati. Questi parametri possono aiutarti a ottimizzare il tuo datastore, verificare che i requisiti del carico di lavoro siano rispettati e fornire una panoramica chiara sull'andamento delle prestazioni del carico di lavoro.

Anti-pattern comuni:

- Utilizzi solo i file di log manuali per la ricerca dei parametri.
- Pubblichi i parametri solo sugli strumenti interni utilizzati dal tuo team e non hai un quadro completo del carico di lavoro.
- Utilizzi solo i parametri predefiniti registrati dal software di monitoraggio selezionato.
- Rivedi i parametri solo quando c'è un problema.
- Monitori solo i parametri a livello di sistema, senza acquisire i parametri di accesso ai dati o di utilizzo.

Vantaggi dell'adozione di questa best practice: la definizione di una linea di base delle prestazioni ti aiuta a comprendere il comportamento normale e i requisiti dei carichi di lavoro. Gli schemi anomali possono essere identificati ed eliminati più rapidamente, per migliorare le prestazioni e l'affidabilità del datastore.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Per monitorare le prestazioni dei datastore, devi registrare più parametri delle prestazioni in un periodo di tempo. Ciò consente di rilevare le anomalie e di misurare le prestazioni rispetto ai parametri aziendali, per verificare che le esigenze del carico di lavoro siano rispettate.

I parametri devono includere sia il sistema sottostante che supporta il datastore sia i parametri del database. I parametri del sistema sottostante possono includere utilizzo della CPU, memoria, spazio di archiviazione su disco disponibile, I/O su disco, percentuale di riscontri nella cache e parametri di rete in entrata e in uscita, mentre i parametri del datastore possono includere transazioni al secondo, query principali, velocità media delle query, tempi di risposta, utilizzo degli indici, blocco delle tabelle, timeout delle query e numero di connessioni aperte. Questi dati sono cruciali per capire l'andamento del carico di lavoro e come viene utilizzata la soluzione di gestione dei dati. Utilizza tali parametri come parte di un approccio basato sui dati per mettere a punto e ottimizzare le risorse del tuo carico di lavoro.

Utilizza strumenti, librerie e sistemi che registrano misure delle prestazioni relative alle prestazioni del database.

Passaggi dell'implementazione

1. Determina i principali parametri delle prestazioni da monitorare per il tuo datastore.
    a. Parametri e dimensioni - Amazon S3
    b. Metriche di monitoraggio per un'istanza di Amazon RDS
    c. Monitoraggio del carico del database con Performance Insights su Amazon RDS
    d. Panoramica del monitoraggio avanzato
    e. Parametri e dimensioni - DynamoDB
    f. Monitoraggio di DynamoDB Accelerator
    g. Monitoraggio di Amazon MemoryDB for Redis con Amazon CloudWatch
    h. Quali parametri è opportuno monitorare?
    i. Monitoraggio delle prestazioni del cluster Amazon Redshift
    j. Parametri e dimensioni - Timestream
    k. Metriche di Amazon CloudWatch per Amazon Aurora
    l. Registrazione e monitoraggio in Amazon Keyspaces (for Apache Cassandra)
    m. Monitoraggio delle risorse di Amazon Neptune

2. Utilizza una soluzione di registrazione e monitoraggio approvata per raccogliere queste metriche. Amazon CloudWatch può raccogliere i parametri per tutte le risorse dell'architettura. Puoi anche raccogliere e pubblicare parametri personalizzati per ottenere parametri aziendali o derivati. Utilizza CloudWatch o soluzioni di terze parti per impostare allarmi che indicano il superamento delle soglie.
3. Verifica se il monitoraggio dei datastore può trarre vantaggio da una soluzione di machine learning che rileva le anomalie delle prestazioni.
   a. Amazon DevOps Guru per Amazon RDS offre visibilità sui problemi di prestazioni e fornisce suggerimenti per le azioni correttive.
4. Configura la conservazione dei dati nella soluzione di monitoraggio e registrazione per soddisfare i tuoi obiettivi operativi e di sicurezza.
   a. Conservazione dei dati predefinita per i parametri CloudWatch
   b. Conservazione dei dati predefinita per CloudWatch Logs

Risorse

Documenti correlati:

- AWS Database Caching (Memorizzazione nella cache del database AWS)
- 10 suggerimenti prestazionali su Amazon Athena
- Best practice con Amazon Aurora
- DynamoDB Accelerator
- Best practice di Amazon DynamoDB
- Best practice di Amazon Redshift Spectrum (Best practice per Amazon Redshift Spectrum)
- Prestazioni di Amazon Redshift
- Database su cloud AWS
- Amazon RDS Performance Insights

Video correlati:

- AWS re:Invent 2022 - Performance monitoring with Amazon RDS and Aurora, featuring Autodesk
- Database Performance Monitoring and Tuning with Amazon DevOps Guru for Amazon RDS
- AWS re:Invent 2023 - What’s new with AWS file storage

- AWS re:Invent 2023 - Dive deep into Amazon DynamoDB
- AWS re:Invent 2023 - Building and optimizing a data lake on Amazon S3
- AWS re:Invent 2023 - What's new with AWS file storage
- AWS re:Invent 2023 - Dive deep into Amazon DynamoDB
- Best Practices for Monitoring Redis Workloads on Amazon ElastiCache

Esempi correlati:

- AWS Dataset Ingestion Metrics Collection Framework (Framework di raccolta dei parametri di ingestione del set di dati AWS)
- Workshop di monitoraggio Amazon RDS
- AWS Purpose Built Databases Workshop

PERF03-BP04 Implementazione di strategie per migliorare le prestazioni delle query nel datastore

Implementa le strategie per ottimizzare i dati e migliorare le query sui dati in modo da consentire una maggiore scalabilità e prestazioni più efficienti per il tuo carico di lavoro.

Anti-pattern comuni:

- Non suddividi i dati in partizioni nel tuo datastore.
- I dati vengono archiviati in un solo formato di file nel tuo datastore.
- Non usi gli indici nel tuo datastore.

Vantaggi dell'adozione di questa best practice: L'ottimizzazione delle prestazioni dei dati e delle query si traduce in maggiore efficienza, costi inferiori e migliore esperienza utente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

L'ottimizzazione dei dati e delle query è un aspetto critico dell'efficienza delle prestazioni in un datastore, poiché influisce sulle prestazioni e sulla reattività dell'intero carico di lavoro cloud. Le query non ottimizzate possono comportare un maggiore utilizzo delle risorse e rallentamenti, riducendo così l'efficienza complessiva di un datastore.

L'ottimizzazione dei dati include diverse tecniche per garantire prestazioni efficienti per l'archiviazione e l'accesso ai dati. Ciò aiuta anche a migliorare le prestazioni delle query in un datastore. Le strategie chiave includono il partizionamento, la compressione e la denormalizzazione dei dati, che contribuiscono a ottimizzare i dati sia per l'archiviazione che per l'accesso.

Passaggi dell'implementazione

- Esamina e analizza le query sui dati critiche che vengono eseguite nel tuo datastore.
- Individua le query lente del tuo datastore e utilizza i piani di query per comprenderne lo stato attuale.
  - Analisi del piano di query in Amazon Redshift
  - Using EXPLAIN and EXPLAIN ANALYZE in Athena
- Implementa le strategie per migliorare le prestazioni delle query. Alcune strategie chiave sono:
  - Usando un formato di file colonnare (come Parquet o ORC).
  - Compressione dei dati nel datastore per ridurre lo spazio di archiviazione e il funzionamento di I/O.
  - Partizionamento dei dati per suddividere i dati in parti più piccole e ridurre i tempi di analisi dei dati.
    - Partizionamento dei dati in Athena
    - Partizioni e distribuzione dei dati
  - L'indicizzazione dei dati sulle colonne comuni della query.
  - Usa le viste materializzate per le domande frequenti.
    - Comprensione delle viste materializzate
    - Creazione di viste materializzate in Amazon Redshift
  - Scegli l'operazione di unione corretta per la query. Quando unisci due tabelle, specifica la tabella più grande sul lato sinistro dell'unione e la tabella più piccola sul lato destro.
  - La soluzione di caching distribuita migliora la latenza e riduce il numero di operazioni di I/O del database.
  - La manutenzione regolare, ad esempio l'esecuzione di statistiche.
- La sperimentazione e i test delle strategie in un ambiente non di produzione.

Risorse

Documenti correlati:

- Best practice con Amazon Aurora
- Prestazioni di Amazon Redshift
- 10 suggerimenti prestazionali su Amazon Athena
- AWS Database Caching (Memorizzazione nella cache del database AWS)
- Best practice per l'implementazione di Amazon ElastiCache
- Partizionamento dei dati in Athena

Video correlati:

- AWS re:Invent 2023 - AWS storage cost-optimization best practices
- AWS re:Invent 2022 - Performance monitoring with Amazon RDS and Aurora, featuring Autodesk
- Optimize Amazon Athena Queries with New Query Analysis Tools

Esempi correlati:

- Amazon S3 Select - Querying data without servers or databases
- AWS Purpose Built Databases Workshop

PERF03-BP05 Implementa modelli di accesso ai dati che utilizzano la memorizzazione nella cache

Implementa modelli di accesso che possano trarre vantaggio dalla memorizzazione dei dati nella cache per il recupero rapido dei dati a cui si accede di frequente.

Anti-pattern comuni:

- Memorizzare nella cache dati che cambiano in maniera frequente.
- Fare affidamento sui dati memorizzati nella cache come se fossero archiviati in modo duraturo e sempre disponibili.
- Non tenere conto della coerenza dei dati memorizzati nella cache.
- Non monitorare l'efficienza dell'implementazione della cache.

Vantaggi dell'adozione di questa best practice: L'archiviazione dei dati in una cache può migliorare la latenza di lettura, la velocità effettiva di lettura, l'esperienza utente e l'efficienza complessiva, oltre a ridurre i costi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Una cache è un componente software o hardware progettato per archiviare dati in modo che le richieste future degli stessi dati possano essere soddisfatte più velocemente o in modo più efficiente. I dati memorizzati in una cache possono essere ricostruiti in caso di perdita, ripetendo un calcolo precedente o recuperandolo da un altro datastore.

La memorizzazione dei dati nella cache può essere una delle strategie più efficaci per migliorare le prestazioni complessive delle applicazioni e ridurre il carico sulle origini dati primarie sottostanti. I dati possono essere memorizzati nella cache a diversi livelli dell'applicazione, ad esempio all'interno dell'applicazione che effettua chiamate remote, operazione nota come memorizzazione nella cache lato client, o utilizzando un servizio secondario veloce per l'archiviazione dei dati, operazione nota come memorizzazione nella cache remota.

Memorizzazione nella cache lato client

Con la memorizzazione nella cache lato client, ogni client (un'applicazione o un servizio che interroga il datastore di backend) può archiviare localmente i risultati delle proprie query uniche per un periodo di tempo specificato. Ciò può ridurre il numero di richieste a un datastore attraverso la rete perché viene controllata prima la cache del client locale. Se questa non contiene risultati, l'applicazione può interrogare il datastore e archiviare tali risultati localmente. Questo modello consente a ciascun client di archiviare i dati nella sede più vicina possibile (il client stesso), garantendo così la latenza più bassa possibile. I client possono inoltre continuare a eseguire query quando il datastore di backend non è disponibile, aumentando la disponibilità dell'intero sistema.

Uno svantaggio di questo approccio è che quando sono coinvolti più client, potrebbero archiviare localmente gli stessi dati memorizzati nella cache. Ciò si traduce in un utilizzo duplicato dell'archiviazione e nell'incoerenza dei dati tra questi client. Può accadere che un client memorizzi nella cache i risultati di una query e un minuto dopo un altro client esegua la stessa query ottenendo un risultato diverso.

Memorizzazione nella cache remota

Per risolvere il problema della duplicazione dei dati tra client, utilizza un servizio esterno veloce o una cache remotaper archiviare i dati sottoposti a query. Anziché controllare un datastore locale, ogni client controllerà la cache remota prima di interrogare il datastore di backend. Questa strategia consente di ottenere risposte più coerenti tra i client, una migliore efficienza dei dati archiviati e un

volume maggiore di dati memorizzati nella cache, perché lo spazio di archiviazione si dimensiona in maniera indipendente dai client.

Lo svantaggio di una cache remota è che l'intero sistema può registrare una latenza più elevata, poiché è necessario un hop di rete aggiuntivo per controllare la cache remota. Per migliorare la latenza, è possibile utilizzare la memorizzazione nella cache lato client insieme alla memorizzazione nella cache remota, eseguendo così una memorizzazione nella cache su più livelli.

Passaggi dell'implementazione

1. Identifica database, API e servizi di rete che potrebbero trarre vantaggio dalla memorizzazione nella cache. I candidati migliori per la memorizzazione nella cache sono i servizi che presentano carichi di lavoro di lettura elevati, un rapporto lettura/scrittura elevato o che sono costosi da dimensionare.
   - Memorizzazione nella cache del database
   - Abilita la memorizzazione nella cache dell'API per migliorare la velocità di risposta
2. Identifica il tipo di strategia di memorizzazione nella cache più adatto al tuo modello di accesso.
   - Strategie di cache
   - Soluzioni di memorizzazione nella cache AWS
3. Seguici Best practice di memorizzazione nella cache per il tuo datastore.
4. Configura una strategia di invalidazione della cache per tutti i dati, ad esempio un TTL (Time-to-live), che permetta di bilanciare attualità dei dati e riduzione della pressione sul datastore di backend.
5. Abilita funzionalità quali tentativi di connessione automatici, backoff esponenziale, timeout lato client e pool di connessioni nel client, se disponibili, che possono migliorare prestazioni e affidabilità.
   - Best practice: client Redis e Amazon ElastiCache for Redis
6. Monitora la percentuale di riscontri nella cache con un obiettivo dell'80% o superiore. Valori inferiori possono indicare una dimensione della cache insufficiente o un modello di accesso che non sfrutta la memorizzazione nella cache.
   - Quali parametri è opportuno monitorare?
   - Best practices for monitoring Redis workloads on Amazon ElastiCache
   - Monitoring best practices with Amazon ElastiCache for Redis using Amazon CloudWatch
7. Implementa la replica dei dati per distribuire il carico delle letture su più istanze e migliorare le prestazioni e la disponibilità di lettura dei dati.

Risorse

Documenti correlati:

- Using the Amazon ElastiCache Well-Architected Lens
- Monitoring best practices with Amazon ElastiCache for Redis using Amazon CloudWatch
- Quali parametri è opportuno monitorare?
- Performance at Scale with Amazon ElastiCache whitepaper
- Sfide e strategie di caching

Video correlati:

- Amazon ElastiCache Learning Path
- Design for success with Amazon ElastiCache best practices
- AWS re:Invent 2020 - Design for success with Amazon ElastiCache best practices
- AWS re:Invent 2023 - [LAUNCH] Introducing Amazon ElastiCache Serverless
- AWS re:Invent 2022 - 5 great ways to reimagine your data layer with Redis
- AWS re:Invent 2021 - Deep dive on Amazon ElastiCache for Redis

Esempi correlati:

- Boosting MySQL database performance with Amazon ElastiCache for Redis

Reti e distribuzione di contenuti

PERF 4. In che modo selezioni e configuri le risorse di rete nel carico di lavoro?

La soluzione di database più efficace per un determinato sistema può variare in base ai requisiti di disponibilità, coerenza, tolleranza della partizione, latenza, durata, scalabilità e capacità di query. Molti sistemi utilizzano diverse soluzioni di database per vari sottosistemi e attivano funzionalità differenti per migliorare le prestazioni. Selezionare la soluzione e le funzionalità del database sbagliate per un sistema può ridurre l'efficienza delle prestazioni.

Best practice

- PERF04-BP01 In che modo la rete influisce sulle prestazioni

- PERF04-BP02 Valuta le funzionalità di rete disponibili
- PERF04-BP03 Scegli la connettività dedicata o la VPN appropriata per il tuo carico di lavoro
- PERF04-BP04 Utilizzo del bilanciamento del carico per distribuire il traffico su più risorse
- PERF04-BP05 Scelta dei protocolli di rete per migliorare le prestazioni
- PERF04-BP06 Scelta della posizione del carico di lavoro in base ai requisiti di rete
- PERF04-BP07 Ottimizzazione della configurazione di rete in base alle metriche

PERF04-BP01 In che modo la rete influisce sulle prestazioni

Analizza e comprendi in che modo le decisioni correlate alla rete influiscono sul carico di lavoro per fornire prestazioni efficienti e una migliore esperienza utente.

Anti-pattern comuni:

- Tutto il traffico passa attraverso i data center esistenti.
- Si instrada tutto il traffico attraverso i firewall centrali anziché utilizzare strumenti di sicurezza di rete nativi del cloud.
- Si effettua il provisioning delle connessioni AWS Direct Connect senza comprendere gli effettivi requisiti di utilizzo.
- Quando si definiscono le soluzioni di rete, non si considerano le caratteristiche del carico di lavoro e l'overhead della crittografia.
- Per le soluzioni di rete nel cloud si utilizzano concetti e strategie on-premise.

Vantaggi dell'adozione di questa best practice: Comprendere l'impatto della rete sulle prestazioni del carico di lavoro ti aiuta a identificare i potenziali colli di bottiglia, migliorare l'esperienza dell'utente, aumentare l'affidabilità e ridurre la manutenzione operativa al variare del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

La rete è responsabile della connettività tra componenti dell'applicazione, servizi cloud, reti edge e dati on-premise e quindi può avere un forte impatto sulle prestazioni dei carichi di lavoro. Oltre alle prestazioni del carico di lavoro, l'esperienza dell'utente può essere influenzata anche da latenza della rete, larghezza di banda, protocolli, posizione, congestione della rete, jitter, velocità di trasmissione effettiva e regole di instradamento.

Disporre di un elenco documentato dei requisiti di rete del carico di lavoro, tra cui latenza, dimensione dei pacchetti, regole di instradamento, protocolli e modelli di traffico di supporto. Esaminare le soluzioni di rete disponibili e individuare il servizio che soddisfi le caratteristiche di rete del proprio carico di lavoro. Le reti basate sul cloud possono essere ricostruite rapidamente, quindi l'evoluzione dell'architettura di rete nel tempo è necessaria per migliorare l'efficienza delle prestazioni.

Passaggi dell'implementazione:

1. Definisci e documenta i requisiti di prestazioni di rete, tra cui metriche come latenza di rete, larghezza di banda, protocolli, posizioni, modelli di traffico (picchi e frequenza), velocità di trasmissione effettiva, crittografia, ispezione e regole di instradamento.
2. Scopri i principali servizi di rete AWS come VPC, AWS Direct Connect, Elastic Load Balancing (ELB)e Amazon Route 53.
3. Acquisisci le seguenti caratteristiche di rete fondamentali:

Caratteristiche	Strumenti e metriche
Caratteristiche fondamentali della rete	• Log di flusso VPC • Log di flusso AWS Transit Gateway • Metriche AWS Transit Gateway • Parametri AWS PrivateLink
Caratteristiche di rete dell'applicazione	• Elastic Fabric Adapter • Metriche AWS App Mesh • Parametri Amazon API Gateway
Caratteristiche della rete edge	• Parametri Amazon CloudFront • Parametri Amazon Route 53 • Metriche AWS Global Accelerator
Caratteristiche della rete ibrida	• Metriche AWS Direct Connect • Metriche AWS Site-to-Site VPN • Metriche AWS Client VPN • Parametri WAN Cloud AWS

Caratteristiche	Strumenti e metriche
Caratteristiche della sicurezza di rete	• Metriche AWS Shield, AWS WAF e AWS Network Firewall
Caratteristiche del tracciamento	• AWS X-Ray • VPC Reachability Analyzer • Network Access Analyzer • Amazon Inspector • Usare il RUM Amazon CloudWatch

4. Esegui il benchmark e testa le prestazioni della rete:
   a. Esegui il benchmark della velocità di trasmissione effettiva della rete, poiché alcuni fattori possono influire sulle prestazioni della rete Amazon EC2 quando le istanze si trovano nello stesso VPC. Misura la larghezza di banda della rete tra le istanze Amazon EC2 Linux nello stesso VPC.
   b. Esegui test di carico per sperimentare soluzioni e opzioni di rete.

Risorse

Documenti correlati:

- Application Load Balancer
- Reti avanzate su Linux
- Reti avanzate su Windows
- Gruppi di collocamento
- Abilitazione delle reti avanzate con Elastic Network Adapter (ENA) sulle istanze Linux
- Network Load Balancer
- Nuovi prodotti di rete con AWS
- Transit Gateway
- Passaggio all'instradamento basato sulla latenza in Amazon Route 53
- Endpoint VPC

Video correlati:

- AWS re:Invent 2023 - AWS networking foundations
- AWS re:Invent 2023 - What can networking do for your application?
- AWS re:Invent 2023 - Advanced VPC designs and new capabilities
- AWS re:Invent 2023 - A developer's guide to cloud networking
- AWS re:Invent 2019 - Connectivity to AWS and hybrid AWS network architectures
- AWS re:Invent 2019 - Optimizing Network Performance for Amazon EC2 Instances
- AWS Summit Online - Improve Global Network Performance for Applications
- AWS re:Invent 2020 - Networking best practices and tips with the Well-Architected Framework
- AWS re:Invent 2020 - AWS networking best practices in large-scale migrations

Esempi correlati:

- AWS Transit Gateway and Scalable Security Solutions
- AWS Networking Workshops
- Hands-on Network Firewall Workshop
- Observing and Diagnosing your Network on AWS
- Finding and addressing Network Misconfigurations on AWS

PERF04-BP02 Valuta le funzionalità di rete disponibili

Valuta le funzionalità di rete nel cloud che possono aumentare le prestazioni. Misura l'impatto di tali funzionalità attraverso test, parametri e analisi. Ad esempio, sfrutta le funzionalità a livello di rete disponibili per ridurre latenza, distanza di rete o jitter.

Anti-pattern comuni:

- Rimani all'interno di una regione perché è lì che si trova fisicamente la tua sede centrale.
- Utilizzi i firewall anziché i gruppi di sicurezza per filtrare il traffico.
- Interrompi TLS per l'ispezione del traffico anziché affidarti a gruppi di sicurezza, policy degli endpoint e altre funzionalità native del cloud.
- Utilizzi solo la segmentazione basata su sottoreti anziché i gruppi di sicurezza.

Vantaggi dell'adozione di questa best practice: la valutazione di tutte le funzionalità e le opzioni del servizio consente di ridurre il costo dell'infrastruttura e l'impegno necessario per mantenere il carico

di lavoro e aumentare l'assetto di sicurezza generale. La struttura portante globale di AWS ti aiuta a fornire ai tuoi clienti la migliore esperienza di rete.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

AWS offre servizi come AWS Global Accelerator e Amazon CloudFront , i quali possono contribuire a migliorare le prestazioni della rete, mentre la maggior parte dei servizi AWS include funzionalità di prodotto (come l'Accelerazione del trasferimento Amazon S3) per ottimizzare il traffico di rete.

Analizza quali opzioni di configurazione relative alla rete sono disponibili e come possono influire sul tuo carico di lavoro. L'ottimizzazione delle prestazioni dipende dalla comprensione del modo in cui queste opzioni interagiscono con l'architettura e dall'impatto che hanno sulle prestazioni misurate e sull'esperienza utente.

Passaggi dell'implementazione

- Crea l'elenco dei componenti del carico di lavoro.
  - Prendi in considerazione l'utilizzo di Cloud AWS WAN per creare, gestire e monitorare la rete dell'organizzazione durante la creazione di una rete globale unificata.
  - Monitora le tue reti globali e principali con le metriche di Amazon CloudWatch Logs. Sfrutta Amazon CloudWatch RUM, che fornisce approfondimenti utili per identificare, comprendere e migliorare l'esperienza digitale degli utenti.
  - Visualizza la latenza di rete aggregata tra Regioni AWS e zone di disponibilità, nonché all'interno di ciascuna zona di disponibilità, utilizzando AWS Network Manager , che ti permette di ottenere informazioni dettagliate sul modo in cui le prestazioni delle applicazioni variano in base alle prestazioni della rete AWS sottostante.
  - Utilizza uno strumento per database di gestione delle configurazioni (CMDB) esistente oppure un servizio come AWS Config per creare un inventario del carico di lavoro e della relativa configurazione.
- Se si tratta di un carico di lavoro esistente, individua e documenta l'analisi di benchmark per le metriche relative alle prestazioni, concentrandoti sui colli di bottiglia e sulle aree da migliorare. Le metriche relative alla rete a livello di prestazioni varieranno a seconda dei requisiti aziendali e delle caratteristiche del carico di lavoro. Come punto di partenza, le seguenti metriche possono essere importanti per la revisione del carico di lavoro: larghezza di banda, latenza, perdita di pacchetti, jitter e ritrasmissioni.

- Se si tratta di un nuovo carico di lavoro, esegui i test di carico per individuare eventuali colli di bottiglia relativi alle prestazioni.
- Per tutti i colli di bottiglia di questo tipo riscontrati, esamina le opzioni di configurazione per le soluzioni in uso per individuare le opportunità di miglioramento delle prestazioni. Consulta le seguenti opzioni e funzionalità di rete fondamentali:

Opportunità di miglioramento	Soluzione
Percorso o instradamenti di rete	Utilizza Network Access Analyzer per identific are percorsi o instradamenti.
Protocolli di rete	Consulta PERF04-BP05 Scelta dei protocolli di rete per migliorare le prestazioni
Topologia di rete	Valuta i compromessi a livello di operazioni e prestazioni tra Peering VPC e AWS Transit Gateway quando si collegano più account. AWS Transit Gateway semplifica il modo in cui interconnetti tutti i VPC, che possono essere distribuiti su migliaia di Account AWS e in reti on-premise. Condividi AWS Transit Gateway tra più account utilizzando la funzionalità AWS Resource Access Manager. Consulta PERF04-BP03 Scegli la connettività dedicata o la VPN appropriata per il tuo carico di lavoro
Servizi di rete	AWS Global Accelerator è un servizio di rete che migliora le prestazioni del traffico degli utenti fino al 60% utilizzando l'infrastruttura di rete globale di AWS. Amazon CloudFront può migliorare le prestazioni della distribuzione dei contenuti del tuo carico di lavoro e la latenza a livello globale.

Opportunità di miglioramento	Soluzione
	Utilizza Lambda@edge per eseguire funzioni di personalizzazione dei contenuti che CloudFront distribuisce più vicino agli utenti, ridurre la latenza e migliorare le prestazioni. Amazon Route 53 offre opzioni di instradamento basato sulla latenza, instradamento basato sulla geolocalizzazione, instradamento basato sulla geoprossimitàe instradamento basato su IP per aiutare a migliorare le prestazioni del carico di lavoro per un pubblico globale. Rivedi il traffico del carico di lavoro e la posizione dell'utente quando il carico di lavoro è distribuito a livello globale per individuare quale opzione di instradamento è in grado di ottimizzare le prestazioni del carico di lavoro.
Funzionalità delle risorse di archiviazione	L'Accelerazione del trasferimento Amazon S3 è una funzione che consente agli utenti esterni di sfruttare i vantaggi delle ottimizzazioni di rete di CloudFront per il caricamento dei dati in Amazon S3. Ciò migliora le caratteristiche di trasferimento di grandi quantità di dati da posizioni remote prive di connettività dedicata al Cloud AWS. I punti di accesso multi-regione in Amazon S3 rappresentano una funzionalità che replica i contenuti in più regioni e semplifica il carico di lavoro fornendo un punto di accesso. Quando viene utilizzato un punto di accesso multi-regione, puoi richiedere o scrivere dati in Amazon S3 con il servizio che identifica il bucket con latenza più bassa.

Opportunità di miglioramento	Soluzione
Funzionalità delle risorse di calcolo	Le interfacce di rete elastiche (ENA) utilizzate da istanze Amazon EC2, container e funzioni Lambda sono limitate in base al flusso. Rivedi i gruppi di collocazione per ottimizzare la velocità di trasmissione effettiva EC2. Per evitare colli di bottiglia a livello di flusso, progetta l'applicazione in modo che utilizzi più flussi. Per monitorare le metriche di rete associate al calcolo e avere maggiore visibilit à su di esse, utilizza le metriche CloudWatch ed ethtool. Il comando `ethtool` è incluso nel driver ENA e permette di utilizzare metriche relative alla rete aggiuntive che possono essere pubblicate come metrica personali zzata in CloudWatch. Gli adattatori elastici di rete (ENA) Amazon offrono un'ulteriore ottimizzazione grazie a una migliore velocità di trasmissione effettiva per le istanze all'interno di un gruppo di collocazione cluster. Elastic Fabric Adapter (EFA) è un'interfaccia di rete per le istanze Amazon EC2 che consente di eseguire carichi di lavoro che richiedono elevati livelli di comunicazioni tra i nodi su vasta scala in AWS. Le istanze ottimizzate per Amazon EBS utilizzano uno stack di configurazione ottimizzato e forniscono un'ulteriore capacità dedicata per incrementare l'I/O di Amazon EBS.

Risorse

Documenti correlati:

- Application Load Balancer
- Reti avanzate su Linux
- Reti avanzate su Windows
- Gruppi di collocamento
- Abilitazione delle reti avanzate con Elastic Network Adapter (ENA) sulle istanze Linux
- Network Load Balancer
- Nuovi prodotti di rete con AWS
- Passaggio all'instradamento basato sulla latenza in Amazon Route 53
- Endpoint VPC
- Log di flusso VPC

Video correlati:

- AWS re:Invent 2023 – Ready for what's next? Designing networks for growth and flexibility
- AWS re:Invent 2023 – Advanced VPC designs and new capabilities
- AWS re:Invent 2023 – A developer's guide to cloud networking
- AWS re:Invent 2022 – Dive deep on AWS networking infrastructure
- AWS re:Invent 2019 – Connectivity to AWS and hybrid AWS network architectures
- AWS re:Invent 2018 – Optimizing Network Performance for Amazon EC2 Instances
- AWS Global Accelerator

Esempi correlati:

- AWS Transit Gateway and Scalable Security Solutions
- AWS Networking Workshops
- Observing and diagnosing your network
- Finding and addressing network misconfigurations on AWS

PERF04-BP03 Scegli la connettività dedicata o la VPN appropriata per il tuo carico di lavoro

Quando hai bisogno di una connettività ibrida per connettere risorse on-premise e cloud, assicurati di avere una larghezza di banda adeguata per soddisfare i tuoi requisiti di prestazione. Fai una stima dei requisiti di larghezza di banda e latenza per il carico di lavoro ibrido. I valori calcolati determineranno le tue esigenze di dimensionamento.

Anti-pattern comuni:

- Valutazione delle soluzioni VPN solo per i tuoi requisiti di crittografia di rete.
- Non vengono valutate opzioni di backup o di connettività ridondante.
- Non è possibile identificare tutti i requisiti del carico di lavoro (esigenze di crittografia, protocollo, larghezza di banda e traffico).

Vantaggi dell'adozione di questa best practice: La selezione e la configurazione di soluzioni di connettività appropriate migliorano l'affidabilità del carico di lavoro e massimizzano le prestazioni. L'identificazione di requisiti del carico di lavoro, la pianificazione anticipata e la valutazione di soluzioni ibride ti permetteranno di ridurre al minimo le costose modifiche alla rete fisica e i costi operativi, migliorando al contempo il time-to-value.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Sviluppa un'architettura di rete ibrida basata sui requisiti di larghezza di banda. AWS Direct Connect ti consente di connettere la tua rete on-premise in modo privato con AWS. È utile quando hai bisogno di larghezza di banda elevata, bassa latenza e di mantenere le prestazioni coerenti. Una connessione VPN permette di connettersi in modo sicuro su Internet. Viene utilizzata quando è necessaria solo una connessione temporanea, quando il costo è un fattore importante o come misura di contingenza in attesa che venga stabilita una connettività di rete fisica resiliente mentre AWS Direct Connect è in uso.

Se i tuoi requisiti di larghezza di banda sono elevati, potresti prendere in considerazione l'utilizzo di più AWS Direct Connect o di servizi di VPN. Il traffico può essere bilanciato in termini di carico tra i servizi, ma il bilanciamento del carico tra AWS Direct Connect e VPN è sconsigliato a causa delle differenze di latenza e larghezza di banda.

Passaggi dell'implementazione

1. Calcola i requisiti di larghezza di banda e latenza delle tue app esistenti.

   a. Per i carichi di lavoro esistenti che vengono spostati in AWS, utilizza i dati raccolti dai sistemi di monitoraggio di rete interni.

   b. Per i carichi di lavoro nuovi o esistenti per i quali non sono disponibili dati di monitoraggio, consulta i proprietari dei prodotti per definire metriche sulle prestazioni adeguate e offrire un'esperienza utente soddisfacente.

2. Scegli una connessione dedicata o una VPN come opzione di connettività. A seconda di tutti i requisiti del carico di lavoro (esigenze di crittografia, larghezza di banda e traffico), puoi scegliere AWS Direct Connect o AWS VPN (o entrambi). Il diagramma seguente può aiutarti a scegliere il tipo di connessione appropriato.

   a. AWS Direct Connect fornisce connettività dedicata all'ambiente AWS da 50 Mbps fino a 100 Gbps, utilizzando connessioni dedicate od ospitate. In questo modo, disporrai di latenza gestita e controllata, nonché di larghezza di banda assegnata, in modo che il carico di lavoro possa connettersi con efficienza ad altri ambienti. Ricorrendo a partner AWS Direct Connect, otterrai connettività end-to-end da più ambienti, per una rete estesa con prestazioni coerenti. AWS permette di dimensionare la larghezza di banda di connessione Direct Connect usando connettività nativa a 100 Gbps, gruppi di aggregazione di collegamenti (LAG, Link Aggregation Group) o instradamento ECMP (Equal-Cost Multipath) con BGP.

   b. AWS Site-to-Site VPN offre un servizio VPN gestito che supporta il protocollo IPSec (Internet Protocol security). Quando viene creata una connessione VPN, ogni connessione include due tunnel per la disponibilità elevata.

3. Consulta la documentazione AWS per scegliere l'opzione di connettività appropriata:

   a. Se decidi di utilizzareAWS Direct Connect, seleziona la larghezza di banda appropriata per la tua connettività.

   b. Se utilizzi una AWS Site-to-Site VPN tra più posizioni per connetterti a una Regione AWS, prova a utilizzare una connessione Site-to-Site VPN accelerata per migliorare le prestazioni della rete.

   c. Se il progetto di rete è costituito da una connessione VPN IPSec tramite AWS Direct Connect, prendi in considerazione l'utilizzo di VPN con indirizzo IP privato per migliorare la sicurezza e ottenere la segmentazione. La VPN sito-sito AWS con indirizzo IP privato viene implementata sull'interfaccia virtuale di transito (VIF).

   d. AWS Direct Connect SiteLink consente di creare connessioni ridondanti e a bassa latenza tra i data center in tutto il mondo inviando dati lungo il percorso più veloce tra sedi di AWS Direct Connect, bypassando Regioni AWS.

4. Convalida la configurazione della connettività prima di eseguire l'implementazione in produzione. Esegui test di sicurezza e prestazioni per assicurarti di soddisfare i requisiti di larghezza di banda, affidabilità, latenza e conformità.
5. Monitora regolarmente le prestazioni e l'utilizzo della connettività e ottimizzali, se necessario.

Diagramma di flusso per le prestazioni deterministiche.

Risorse

Documenti correlati:

- Nuovi prodotti di rete con AWS
- AWS Transit Gateway
- Endpoint VPC
- Creazione di un'infrastruttura di rete AWS scalabile e sicura con più VPC
- Client VPN

Video correlati:

- AWS re:Invent 2023 – Building hybrid network connectivity with AWS
- AWS re:Invent 2023 – Secure remote connectivity to AWS
- AWS re:Invent 2022 – Optimizing performance with Amazon CloudFront
- AWS re:Invent 2019 – Connectivity to AWS and hybrid AWS network architectures
- AWS re:Invent 2020 – AWS Transit Gateway Connect

Esempi correlati:

- AWS Transit Gateway and Scalable Security Solutions
- AWS Networking Workshops

PERF04-BP04 Utilizzo del bilanciamento del carico per distribuire il traffico su più risorse

Distribuisci il traffico tra varie risorse o servizi affinché il carico di lavoro possa trarre vantaggio dall'elasticità fornita dal cloud. Puoi anche utilizzare il bilanciamento del carico per la terminazione dell'offloading della crittografia al fine di migliorare le prestazioni, l'affidabilità e gestire e instradare il traffico in modo efficiente.

Anti-pattern comuni:

- Scelta del tipo di sistema di bilanciamento del carico senza tenere conto dei requisiti del carico di lavoro.
- Mancato utilizzo delle funzionalità del sistema di bilanciamento del carico per l'ottimizzazione delle prestazioni.
- Esposizione diretta del carico di lavoro a Internet senza un sistema di bilanciamento del carico.
- Instradi tutto il traffico Internet attraverso i sistemi di bilanciamento del carico esistenti.
- Utilizzi il bilanciamento del carico TCP generico e fai in modo che ogni nodo di calcolo gestisca la crittografia SSL.

Vantaggi dell'adozione di questa best practice: un bilanciatore del carico gestisce il carico variabile del traffico dell'applicazione in una o più zone di disponibilità e fornisce alta disponibilità, dimensionamento automatico e un migliore utilizzo del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I sistemi di bilanciamento del carico operano come punto di ingresso per il carico di lavoro, dal quale distribuiscono il traffico alle destinazioni di backend, come istanze di calcolo o container per migliorarne l'utilizzo.

La scelta del tipo corretto di sistema di bilanciamento del carico è il primo passaggio per ottimizzare l'architettura. Per iniziare, elenca le caratteristiche del carico di lavoro, tra cui protocollo (TCP, HTTP, TLS o WebSocket), tipo di destinazione (istanze, container o servizi serverless), requisiti dell'applicazione (connessioni a esecuzione prolungata, autenticazione utente o persistenza) e ubicazione (regione, zona locale, Outpost o isolamento zonale).

AWS fornisce più modelli per consentire alle tue applicazioni di utilizzare il bilanciamento del carico. Application Load Balancer è l'ideale per il bilanciamento del carico del traffico HTTP e HTTPS,

nonché offre l'instradamento avanzato delle richieste, dedicato alla distribuzione delle architetture applicative moderne, fra cui microservizi e container.

Network Load Balancer è l'ideale per il bilanciamento del carico del traffico TCP, in cui sono richieste prestazioni elevatissime. È in grado di gestire milioni di richieste al secondo, mantenendo al contempo latenze ridottissime. Inoltre, è ottimizzato per la gestione degli schemi di traffico improvvisi e incostanti.

Elastic Load Balancing fornisce la gestione integrata dei certificati e la decrittografia SSL/TLS, offrendoti la flessibilità di gestire centralmente le impostazioni SSL del bilanciatore del carico e di sollevare il carico di lavoro dall'utilizzo intensivo della CPU.

Dopo aver scelto il sistema di bilanciamento del carico appropriato, puoi iniziare a utilizzarne le funzionalità per ridurre la quantità di attività che deve svolgere il backend per distribuire il traffico.

Ad esempio, usando un Application Load Balancer (ALB) e un Network Load Balancer (NLB), puoi eseguire l'offload della crittografia SSL/TLS, il che costituisce un'opportunità per evitare il completamento dell'handshake TLS a elevato utilizzo di CPU da parte delle destinazioni e migliorare anche la gestione dei certificati.

Se configurato nel sistema di bilanciamento del carico, l'offload SSL/TLS diventa responsabile della crittografia del traffico da e verso i client, distribuendo il traffico non crittografato ai backend, liberando le risorse backend e migliorando il tempo di risposta per i client.

L'Application Load Balancer può anche distribuire traffico HTTP/2 senza che questo debba essere supportato nelle destinazioni. Questa semplice decisione può migliorare il tempo di risposta dell'applicazione, in quanto HTTP/2 usa connessioni TCP in modo più efficiente.

Nel definire l'architettura, è bene tenere conto dei requisiti di latenza del carico di lavoro. Ad esempio, se un'applicazione è sensibile alla latenza, è possibile scegliere di usare un Network Load Balancer, che offre latenze estremamente ridotte. In alternativa, è possibile decidere di avvicinare il carico di lavoro ai clienti utilizzando Application Load Balancer in zone locali AWS o anche in AWS Outposts.

Un altro aspetto di cui tenere conto per i carichi di lavoro sensibili alla latenza è il bilanciamento del carico tra zone. Con il bilanciamento del carico tra zone, ogni nodo del sistema di bilanciamento del carico distribuisce il traffico tra le destinazioni registrate in tutte le zone di disponibilità autorizzate.

Usa Auto Scaling integrato con il sistema di bilanciamento del carico. Uno degli aspetti principali di un sistema con prestazioni efficienti riguarda il dimensionamento corretto delle risorse backend.

A questo scopo, puoi utilizzare integrazioni dei sistemi di bilanciamento del carico per le risorse di destinazione backend. Usando l'integrazione dei sistemi di bilanciamento del carico con gruppi con Auto Scaling, le destinazioni vengono aggiunte o rimosse nel e dal sistema di bilanciamento del carico in base alle esigenze, in risposta al traffico in ingresso. I bilanciatori del carico possono integrarsi anche con Amazon ECS e Amazon EKS per i carichi di lavoro distribuiti in container.

- Amazon ECS - Service load balancing
- Application load balancing on Amazon EKS
- Network load balancing on Amazon EKS

Passaggi dell'implementazione

- Definisci i tuoi requisiti di bilanciamento del carico, tra cui volume di traffico, disponibilità e scalabilità delle applicazioni.
- Scegli il tipo di sistema di bilanciamento del carico giusto per la tua applicazione.
  - Usa un Application Load Balancer per carichi di lavoro HTTP/HTTPS.
  - Use un Network Load Balancer per carichi di lavoro non HTTP in esecuzione su TCP o UDP.
  - Usa una combinazione dei due sistemi (un ALB come destinazione di un NLB) se vuoi usufruire delle funzionalità di entrambi i prodotti. Ad esempio, puoi scegliere questa opzione per usare gli indirizzi IP statici del NLB insieme all'instradamento basato su intestazione HTTP dell'ALB oppure se vuoi esporre il carico di lavoro HTTP a un AWS PrivateLink.
  - Per un confronto completo dei bilanciatori del carico, consulta la tabella di confronto dei prodotti ELB.
- Se possibile, utilizza l'offload SSL/TLS.
  - Configura gli ascoltatori HTTPS/TLS con un Application Load Balancer e un Network Load Balancer integrati con AWS Certificate Manager.
  - Alcuni carichi di lavoro possono richiedere la crittografia end-to-end per motivi di conformità. In questo caso, è necessario consentire la crittografia nelle destinazioni.
  - Per le best practice per la sicurezza, consulta SEC09-BP02 Applicazione della crittografia dei dati in transito.
- Seleziona l'algoritmo di instradamento corretto (solo ALB)
  - L'algoritmo di instradamento può fare la differenza per quanto riguarda l'uso corretto delle destinazioni backend e, di conseguenza, l'impatto sulle prestazioni. Ad esempio, l'ALB offre due opzioni per gli algoritmi di instradamento:

  - Numero minimo di richieste in sospeso: usa questa opzione per ottenere una migliore distribuzione del carico nelle destinazioni back-end nei casi in cui le richieste per l'applicazione variano per complessità o le destinazioni variano per capacità di elaborazione.
  - Round robin: usa questa opzione quando le richieste e le destinazioni sono simili o se devi distribuire equamente le richieste tra le destinazioni.
- Valuta se usare l'isolamento tra zone o quello zonale.
  - Disattiva l'isolamento tra zone (usando l'isolamento zonale) per migliorare la latenza e in caso di errori di zona. È disattivato per impostazione predefinita nel NLB, mentre nell'ALB puoi disattivarlo per ogni gruppo di destinazioni.
  - Attiva l'isolamento tra zone per ottenere disponibilità e flessibilità maggiori. Per impostazione predefinita, l'isolamento tra zone è disattivato per l'ALB, mentre nel NLB puoi attivarlo per ogni gruppo di destinazioni.
- Attiva keep-alive HTTP per i carichi di lavoro HTTP (solo ALB). Con questa funzionalità, il sistema di bilanciamento del carico può riutilizzare le connessioni backend fino allo scadere del timeout del keep-alive, migliorando la richiesta HTTP e il tempo di risposta e riducendo anche l'utilizzo delle risorse nelle destinazioni backend. Per informazioni sulla configurazione per Apache e Nginx, consulta Quali sono le impostazioni ottimali per utilizzare Apache o NGINX come server di backend per ELB?
- Attiva il monitoraggio del tuo sistema di bilanciamento del carico.
  - Attiva i log di accesso per l'Application Load Balancer e il Network Load Balancer.
  - I campi principali da considerare per l'ALB sono `request_processing_time`, `request_processing_time` e `response_processing_time`.
  - I campi principali da considerare per il NLB sono `connection_time` e `tls_handshake_time`.
  - Preparati a eseguire query sui log quando necessario. Puoi usare Amazon Athena per eseguire query su log dell'ALB e log del NLB.
  - Crea gli allarmi per le metriche relative alle prestazioni come `TargetResponseTime` per l'ALB.

Risorse

Documenti correlati:

- Confronti di prodotti ELB
- Infrastruttura globale di AWS

- Improving Performance and Reducing Cost Using Availability Zone Affinity
- Step by step for Log Analysis with Amazon Athena
- Querying Application Load Balancer logs
- Monitor your Application Load Balancers
- Monitor your Network Load Balancer
- Use Elastic Load Balancing to distribute traffic across the instances in your Auto Scaling group

Video correlati:

- AWS re:Invent 2023: What can networking do for your application?
- AWS re:Inforce 20: How to use Elastic Load Balancing to enhance your security posture at scale
- AWS re:Invent 2018: Elastic Load Balancing: Deep Dive and Best Practices
- AWS re:Invent 2021 - How to choose the right load balancer for your AWS workloads
- AWS re:Invent 2019: Get the most from Elastic Load Balancing for different workloads

Esempi correlati:

- Gateway Load Balancer
- CDK and AWS CloudFormation samples for Log Analysis with Amazon Athena

PERF04-BP05 Scelta dei protocolli di rete per migliorare le prestazioni

Prendi decisioni sui protocolli per la comunicazione tra sistemi e reti in base all'impatto sulle prestazioni del carico di lavoro.

Esiste una relazione tra latenza e larghezza di banda per ottenere la velocità di trasmissione desiderata. Se per il trasferimento di file viene usato il protocollo TCP, latenze più elevate molto probabilmente ridurranno la velocità di trasmissione effettiva complessiva. Alcuni approcci risolvono questo problema tramite l'ottimizzazione del TCP e l'utilizzo di protocolli di trasferimento ottimizzati, un altro prevede l'utilizzo del protocollo UDP (User Datagram Protocol).

Anti-pattern comuni:

- Puoi utilizzare il TCP per tutti i carichi di lavoro, indipendentemente dai requisiti prestazionali.

Vantaggi dell'adozione di questa best practice: La verifica del protocollo appropriato per la comunicazione tra utenti e componenti del carico di lavoro contribuisce a migliorare l'esperienza utente complessiva per le applicazioni. Ad esempio, l'UDP senza connessione garantisce velocità elevata, ma non offre ritrasmissione o alta affidabilità. Il TCP è un protocollo completo, ma richiede un sovraccarico maggiore per l'elaborazione dei pacchetti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Se hai la possibilità di scegliere protocolli diversi per la tua applicazione e hai esperienza in questo campo, ottimizza l'applicazione e l'esperienza dell'utente finale utilizzando un protocollo diverso. Tieni conto che questo approccio presenta notevoli difficoltà e dovrebbe essere tentato solo dopo aver ottimizzato l'applicazione in altri modi.

Un aspetto principale per il miglioramento delle prestazioni del carico di lavoro consiste nell'identificare i requisiti di latenza e velocità di trasmissione effettiva e quindi scegliere i protocolli di rete che ottimizzano le prestazioni.

Quando valutare se usare TCP

TCP permette la trasmissione affidabile dei dati e può essere usato per la comunicazione tra i componenti del carico di lavoro quando l'affidabilità e la garanzia di trasmissione dei dati sono due aspetti importanti. Molte applicazioni Web usano protocolli basati su TCP, come HTTP e HTTPS, per aprire socket TCP per la comunicazione tra i componenti dell'applicazione. Il TCP viene comunemente usato per il trasferimento di dati di posta elettronica e di file, in quanto è un meccanismo di trasferimento semplice e affidabile tra i componenti dell'applicazione. L'uso di TLS con TCP può aggiungere un certo sovraccarico alla comunicazione, il che produce maggiore latenza e velocità di trasmissione effettiva inferiore, ma presenta come vantaggio una maggiore sicurezza. Il sovraccarico è dovuto prevalentemente al processo di handshake, il cui completamento può richiedere diversi round trip. Al termine del processo di handshake, il sovraccarico dovuto alla crittografia e alla decrittografia dei dati è relativamente ridotto.

Quando valutare se usare UDP

UDP è un protocollo di tipo connection-less (senza connessione) e di conseguenza è ideale per applicazioni che necessitano di una trasmissione veloce ed efficiente, ad esempio per i log, il monitoraggio e i dati VoIP. Valuta se usare UDP anche se vi sono componenti del carico di lavoro che rispondono a piccole query provenienti da grandi quantità di client per garantire prestazioni

ottimali del carico di lavoro. Datagram Transport Layer Security (DTLS) è l'equivalente UDP di Transport Layer Security (TLS). Quando viene usato DTLS con UDP, il sovraccarico è dovuto alla crittografia e alla decrittografia dei dati, in quanto il processo di handshake è semplificato. DTLS aggiunge anche un piccolo sovraccarico ai pacchetti UDP, perché include altri campi per indicare i parametri di sicurezza e rilevare la manomissione.

Quando valutare se usare SRD

SRD (Scalable Reliable Datagram) è un protocollo di trasporto di rete ottimizzato per carichi di lavoro a velocità di trasmissione effettiva elevata grazie alla sua capacità di bilanciare il carico del traffico tra più percorsi e di recuperare rapidamente dalla perdita di pacchetti e da errori di collegamento. Di conseguenza, SRD è ideale per carichi di lavoro di calcolo ad alte prestazioni (HPC) che richiedono comunicazioni tra nodi di calcolo a velocità di trasmissione effettiva elevata e a bassa latenza. Possono essere incluse attività di elaborazione in parallelo come la simulazione, la modellazione e l'analisi dei dati che implicano il trasferimento di grandi quantità di dati tra nodi.

Passaggi dell'implementazione

1. Utilizza AWS Global Accelerator e AWS Transfer Family per migliorare la velocità di trasmissione effettiva delle applicazioni di trasferimento di file online. Il servizio AWS Global Accelerator ti permette di ottenere latenza inferiore tra i dispositivi client e il carico di lavoro in AWS. Con AWS Transfer Family puoi usare protocolli basati su TCP come SFTP (Secure Shell File Transfer Protocol) ed FTPS (File Transfer Protocol over SSL) per dimensionare e gestire i trasferimenti di file in servizi di archiviazione AWS in tutta sicurezza.
2. Usa la latenza di rete per determinare se TCP sia il protocollo appropriato per la comunicazione tra componenti del carico di lavoro. Se la latenza di rete tra l'applicazione client e il server è elevata, il processo di handshake a tre vie tramite TCP può richiedere tempo, influendo sulla velocità di risposta dell'applicazione. Per misurare la latenza di rete, puoi usare, ad esempio, le metriche TTFB (tempo di acquisizione al primo byte) e RTT (tempo di andata e ritorno). Se il tuo carico di lavoro fornisce agli utenti contenuti dinamici, prendi in considerazione l'utilizzo di Amazon CloudFront, che stabilisce una connessione persistente a ogni origine per il contenuto dinamico in modo da eliminare il tempo di configurazione della connessione, che altrimenti rallenterebbe ogni richiesta client.
3. L'uso di TLS con TCP o UDP può causare maggiore latenza e minore velocità di trasmissione effettiva per il carico di lavoro a causa dell'impatto della crittografia e della decrittografia. Per carichi di lavoro di questo tipo, prendi in considerazione l'offload SSL/TLS in Elastic Load Balancing per migliorare le prestazioni permettendo al sistema di bilanciamento del carico di gestire la crittografia e la decrittografia SSL/TLS invece di predisporre a questo scopo istanze

back-end. In questo modo, puoi ridurre l'utilizzo della CPU sulle istanze back-end, migliorando le prestazioni e aumentando la capacità.

4. Utilizza il Network Load Balancer (NLB) per implementare servizi basati sul protocollo UDP, tra cui autenticazione e autorizzazione, registrazione, DNS, IoT e streaming di contenuti multimediali, in modo da migliorare le prestazioni e l'affidabilità del carico di lavoro. L'NLB distribuisce il traffico UDP in ingresso tra più destinazioni, permettendo di aumentare o ridurre orizzontalmente il carico di lavoro, incrementare la capacità e diminuire il sovraccarico su un'unica destinazione.
5. Per i tuoi carichi di lavoro HPC (calcolo ad alte prestazioni), prendi in considerazione l'utilizzo della funzionalità Adattatore elastico di rete (ENA) Express , che usa il protocollo SRD per migliorare le prestazioni di rete fornendo una larghezza di banda a flusso singolo più elevata (25 Gbps) e una latenza di coda inferiore (99,9 percentile) per il traffico di rete tra istanze EC2.
6. Utilizza l'Application Load Balancer (ALB) per instradare il traffico gRPC (Remote Procedure Call) tra componenti del carico di lavoro o tra client e servizi gRPC e per bilanciarne il carico. gRPC usa il protocollo HTTP/2 basato su TCP per il trasporto e fornisce vantaggi in termini di prestazioni, tra cui un impatto di rete minore, la compressione, la serializzazione binaria efficiente, il supporto per diversi linguaggi e lo streaming bidirezionale.

Risorse

Documenti correlati:

- How to route UDP traffic into Kubernetes
- Application Load Balancer
- Reti avanzate su Linux
- Reti avanzate su Windows
- Gruppi di collocamento
- Abilitazione delle reti avanzate con Elastic Network Adapter (ENA) sulle istanze Linux
- Network Load Balancer
- Nuovi prodotti di rete con AWS
- Passaggio all'instradamento basato sulla latenza in Amazon Route 53
- Endpoint VPC

Video correlati:

- AWS re:Invent 2022 – Scaling network performance on next-gen Amazon Elastic Compute Cloud instances
- AWS re:Invent 2022 – Application networking foundations

Esempi correlati:

- AWS Transit Gateway and Scalable Security Solutions
- AWS Networking Workshops

PERF04-BP06 Scelta della posizione del carico di lavoro in base ai requisiti di rete

Valuta le opzioni per il posizionamento delle risorse in modo da diminuire la latenza di rete e migliorare la velocità di trasmissione effettiva, fornendo un'esperienza utente ottimale attraverso la riduzione dei tempi di caricamento delle pagine e di trasferimento dei dati.

Anti-pattern comuni:

- Consolidamento di tutte le risorse del carico di lavoro in un'unica posizione geografica.
- Scelta della regione più vicina alla propria posizione, ma non al carico di lavoro dell'utente finale.

Vantaggi dell'adozione di questa best practice: l'esperienza utente è fortemente condizionata dalla latenza tra utente e applicazione. Utilizzando le Regioni AWS appropriate e una rete globale AWS privata, puoi ridurre la latenza e offrire un'esperienza migliore agli utenti remoti.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Le risorse, ad esempio le istanze Amazon EC2, vengono posizionate in zone di disponibilità all'interno delle Regioni AWS, in zone locali AWS, in AWS Outposts o in zone AWS Wavelength. La scelta della posizione influisce sulla latenza di rete e sulla velocità di trasmissione effettiva dall'ubicazione di un utente specifico. È anche possibile usare i servizi edge, quali Amazon CloudFront e AWS Global Accelerator, per migliorare le prestazioni della rete memorizzando i contenuti nella cache delle posizioni edge o fornendo agli utenti il percorso ottimale del carico di lavoro tramite la rete globale AWS.

Amazon EC2 offre gruppi di collocazione per le reti. Un gruppo di collocazione è un raggruppamento logico di istanze per ridurre la latenza. L'utilizzo di gruppi di collocazione con tipi di istanza supportati

e un Adattatore elastico di rete (ENA) consente ai carichi di lavoro di partecipare a una rete a 25 Gbps a bassa latenza e a jitter ridotto. I gruppi di collocazione sono consigliati per i carichi di lavoro che traggono beneficio da reti a bassa latenza, throughput di rete elevato o entrambi.

I servizi sensibili alla latenza vengono forniti nelle posizioni edge utilizzando una rete AWS globale, ad esempio Amazon CloudFront. Tali posizioni edge forniscono solitamente servizi come rete di distribuzione di contenuti (CDN) e sistema dei nomi di dominio (DNS). Fornendo questi servizi nell'edge, possono rispondere con una latenza ridotta alle richieste di contenuti o risoluzione DNS. Inoltre, possono offrire servizi geografici come la geotargetizzazione dei contenuti (ossia fornire contenuti diversi in base alla posizione dell'utente finale) o l'instradamento basato sulla latenza, per indirizzare gli utenti alla regione più vicina (latenza minima).

Usa i servizi edge per ridurre la latenza e abilitare la memorizzazione nella cache dei contenuti. Configura correttamente il controllo cache sia per DNS sia per HTTP/HTTPS al fine di sfruttare tutti i vantaggi offerti da tali approcci.

Passaggi dell'implementazione

- Acquisisci informazioni sul traffico IP in entrata e in uscita dalle interfacce di rete.
  - Log del traffico IP tramite log di flusso VPC
  - Come viene conservato l’indirizzo IP del client in AWS Global Accelerator
- Analizza i modelli di accesso alla rete nel tuo carico di lavoro per capire come gli utenti usano la tua applicazione.
  - Usa strumenti di monitoraggio, come Amazon CloudWatch e AWS CloudTrail, per raccogliere i dati sull'attività della rete.
  - Analizza i dati per identificare il modello di accesso alla rete.
- Seleziona regioni appropriate per l'implementazione del carico di lavoro in base ai seguenti elementi chiave:
  - Dove si trovano i tuoi dati: per le applicazioni a uso intensivo di dati, ad esempio applicazioni di big data e machine learning, il codice dell'applicazione deve essere eseguito il più vicino possibile ai dati.
  - Dove si trovano i tuoi utenti: per le applicazioni rivolte agli utenti, scegli una o più regioni vicine agli utenti del tuo carico di lavoro.
  - Altre limitazioni: considera le limitazioni relative a costi e conformità, come spiegato nel post What to Consider when Selecting a Region for your Workloads..

- Usa le zone locali AWS per eseguire carichi di lavoro come il rendering di video. Le zone locali consentono di sfruttare i vantaggi derivanti dalla disponibilità di risorse di calcolo e archiviazione più vicine agli utenti finali.
- Usa AWS Outposts per carichi di lavoro che devono rimanere in locale, ma vuoi che vengano eseguiti in modo ottimale con il resto degli altri carichi di lavoro in AWS.
- Applicazioni come quelle di streaming di video live ad alta risoluzione, audio ad alta fedeltà o realtà aumentata o realtà virtuale (AR/VR) richiedono latenza bassissima per i dispositivi 5G. Per applicazioni di questo tipo, prendi in considerazione AWS Wavelength. AWS Wavelength incorpora servizi di calcolo e archiviazione AWS in reti 5G, fornendo un'infrastruttura di edge computing per dispositivi mobili per lo sviluppo, l'implementazione e il dimensionamento di applicazioni a latenza bassissima.
- Usa la cache locale o le Soluzioni per la acche di AWS per le risorse di frequente utilizzo per migliorare le performance, ridurre lo spostamento dei dati e minimizzare l'impatto ambientale.

Service	When to use
Amazon CloudFront	Usa per memorizzare nella cache contenuti statici come immagini, script e video, nonché contenuti dinamici come risposte API o applicazioni Web.
Amazon ElastiCache	Usa per memorizzare nella cache i contenuti per le applicazioni Web.
DynamoDB Accelerator	Usa per aggiungere accelerazione in memoria alle tabelle DynamoDB.

- Utilizza servizi in grado di supportarti nell'esecuzione del codice in posizioni più vicine agli utenti del carico di lavoro, come i seguenti:

Service	When to use
Lambda@edge	Usa per operazioni a uso intensivo di risorse di calcolo eseguite quando gli oggetti non si trovano nella cache.

Service	When to use
Funzioni Amazon CloudFront	Usa per casi d'uso semplici, ad esempio manipolazioni di risposte o richieste HTTP(s) che possono essere avviate da funzioni di breve durata.
AWS IoT Greengrass	Usa per eseguire la memorizzazione nella cache di risorse di calcolo, messaggistica e dati per i dispositivi connessi.

- Alcune applicazioni richiedono punti di ingresso fissi o prestazioni più elevate attraverso la riduzione della latenza di ricezione del primo byte e l'instabilità e l'aumento della velocità di trasmissione effettiva. Queste applicazioni possono trarre vantaggio da servizi di rete che forniscono indirizzi IP anycast statici e terminazione TCP in posizioni edge. AWS Global Accelerator può migliorare le prestazioni per le applicazioni fino al 60% e offre un failover rapido per architetture in più regioni. AWS Global Accelerator fornisce indirizzi IP anycast statici che fungono da punto di ingresso fisso per le applicazioni ospitate in una o più Regioni AWS. Questi indirizzi IP permettono l'ingresso del traffico nella rete AWS globale più vicina possibile agli utenti. AWS Global Accelerator riduce il tempo di configurazione della connessione iniziale stabilendo una connessione TCP tra il client e la posizione edge di AWS più vicina al client. Riesamina l'uso di AWS Global Accelerator per migliorare le prestazioni dei carichi di lavoro TCP/UDP e fornire il rapido failover per architetture in più regioni.

Risorse

Best practice correlate:

- COST07-BP02 Implementazione delle regioni in base al costo
- COST08-BP03 Implementazione dei servizi per ridurre il costo di trasferimento dei dati
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL10-BP02 Selezione delle posizioni appropriate per la tua implementazione multiposizione
- SUS01-BP01 Scelta della Regione in base alle esigenze aziendali e agli obiettivi di sostenibilità.
- SUS02-BP04 Ottimizzazione del posizionamento geografico dei carichi di lavoro in base ai requisiti di rete
- SUS04-BP07 Riduzione al minimo dello spostamento di dati tra reti

Documenti correlati:

- Infrastruttura globale AWS
- Zone locali AWS e AWS Outposts, scelta della giusta tecnologia per un carico di lavoro edge
- Gruppi di collocazione
- Zone locali AWS
- AWS Outposts
- AWS Wavelength
- Amazon CloudFront
- AWS Global Accelerator
- AWS Direct Connect
- AWS Site-to-Site VPN
- Amazon Route 53

Video correlati:

- Video di presentazione delle zone locali AWS
- AWS Outposts: Overview and How it Works
- AWS re:Invent 2023 - A migration strategy for edge and on-premises workloads
- AWS re:Invent 2021: AWS Outposts: Spostamento dell'esperienza AWS in un ambiente on-premise
- AWS re:Invent 2020: AWS Wavelength: esecuzione di app con latenza bassissima nell'edge 5G
- AWS re:Invent 2022: Zone locali AWS: creazione di applicazioni per una posizione edge distribuita
- AWS re:Invent 2021: Creazione di siti Web a bassa latenza con Amazon CloudFront
- AWS re:Invent 2022: Miglioramento delle prestazioni e della disponibilità con AWS Global Accelerator
- AWS re:Invent 2022: Creazione di una rete WAN usando AWS
- AWS re:Invent 2020: Gestione del traffico globale con Amazon Route 53

Esempi correlati:

- AWS Global Accelerator Custom Routing Workshop
- Gestione delle riscritture e dei reindirizzamenti usando funzioni di edge computing

PERF04-BP07 Ottimizzazione della configurazione di rete in base alle metriche

Usa i dati raccolti e analizzati per prendere decisioni informate riguardo l'ottimizzazione della configurazione della tua rete.

Anti-pattern comuni:

- Ritieni che tutti i problemi relativi alle prestazioni siano correlati all'applicazione.
- Verifica delle prestazioni di rete solo da una posizione vicina a quella in cui è stato distribuito il carico di lavoro.
- Uso di configurazioni predefinite per tutti i servizi di rete.
- Provisioning in eccesso di risorse di rete per fornire capacità sufficiente.

Vantaggi dell'adozione di questa best practice: la raccolta delle metriche necessarie per la rete AWS e l'implementazione di strumenti di monitoraggio di rete permettono di identificare le prestazioni di rete e ottimizzare le configurazioni di rete.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Il monitoraggio del traffico da e verso VPC, sottoreti o interfacce di rete è essenziale per identificare come utilizzare risorse di rete AWS e ottimizzare le configurazioni di rete. Usando i seguenti strumenti di rete AWS, puoi esaminare ulteriormente le informazioni sull'utilizzo del traffico, sull'accesso alla rete e sui log.

Passaggi dell'implementazione

- Identifica le metriche delle prestazioni fondamentali da raccogliere, come la latenza o la perdita di pacchetti. AWS fornisce diversi strumenti che possono aiutarti a raccogliere queste metriche. Usando i seguenti strumenti, puoi esaminare ulteriormente le informazioni sull'utilizzo del traffico, sull'accesso alla rete e sui log:

Strumento AWS	Dove usare
Amazon VPC IP Address Manager.	Utilizza IPAM per pianificare, seguire e monitorare gli indirizzi IP per i carichi di lavoro AWS e on-premise. Si tratta di una best

Strumento AWS	Dove usare
	practice per ottimizzare l'utilizzo e l'allocazione degli indirizzi IP.
Log di flusso VPC	Usa log di flusso VPC per acquisire informazioni dettagliate sul traffico da e verso le interfacce di rete nei VPC. Con i log di flusso VPC puoi diagnosticare regole dei gruppi di sicurezza eccessivamente restrittive o permissive e determinare la direzione del traffico da e verso le interfacce di rete.
Log di flusso AWS Transit Gateway	Utilizza i log di flusso AWS Transit Gateway per acquisire informazioni sul traffico IP in entrata e in uscita dai gateway di transito.
Registrazione di query DNS	Registra le informazioni sulle query DNS pubbliche o private ricevute da Route 53. Con i log DNS puoi ottimizzare le configurazioni DNS identificando il dominio e il sottodomi nio richiesto o le posizioni edge Route 53 che hanno risposto a query DNS.
Reachability Analyzer	Reachability Analyzer ti aiuta a effettuare l'analisi e il debug della raggiungibilità della rete. Reachability Analyzer è uno strumento di analisi della configurazione che permette di eseguire test di connettività tra una risorsa di origine e una risorsa di destinazione nei VPC. Questo strumento permette di verificare che la configurazione di rete corrisponda alla connettività desiderata.

Strumento AWS	Dove usare
Network Access Analyzer	Network Access Analyzer ti aiuta a definire l'accesso alla rete per le tue risorse. Puoi usare Network Access Analyzer per specifica re i requisiti di accesso alla rete e identificare i potenziali percorsi di rete che non li soddisfan o. Ottimizzando la configurazione di rete corrispondente, puoi determinare e verificare lo stato della rete e indicare se la rete su AWS soddisfa i requisiti di conformità.
Amazon CloudWatch	Utilizza Amazon CloudWatch e attiva le metriche appropriate per le opzioni di rete. Assicurati di scegliere le metriche di rete corrette per il carico di lavoro. Ad esempio, puoi attivare le metriche per l'utilizzo degli indirizzi di rete del VPC, il gateway NAT del VPC, AWS Transit Gateway, il tunnel VPN, AWS Network Firewall, Elastic Load Balancing e AWS Direct Connect. Il monitorag gio continuo delle metriche è una procedura utile per osservare e identificare lo stato e l'utilizzo della rete che semplifica l'ottimiz zazione della configurazione di rete in base alle osservazioni.
AWS Network Manager	Con AWS Network Manager puoi monitorar e le prestazioni in tempo reale e storiche della rete globale AWS per scopi operativi e di pianificazione. Network Manager fornisce una latenza di rete aggregata tra Regioni AWS e zone di disponibilità e all'interno di ciascuna zona di disponibilità, permettendoti di comprendere meglio in che modo le prestazio ni delle applicazioni si relazionano con le prestazioni della rete AWS sottostante.

Strumento AWS	Dove usare
Amazon CloudWatch RUM	Usa Amazon CloudWatch RUM per raccogliere le metriche che ti consentono di ottenere approfondimenti utili per identificare, comprendere e migliorare l'esperienza utente.

- Identifica i top talker e gli schemi di traffico delle applicazioni utilizzando VPC e i log di flusso di AWS Transit Gateway.
- Valuta e ottimizza la tua attuale architettura di rete, inclusi VPC, sottoreti e routing. Ad esempio, puoi valutare come i diversi VPC per il peering o AWS Transit Gateway possono aiutarti a migliorare la rete nella tua architettura.
- Valuta i percorsi di routing nella tua rete per verificare che venga sempre utilizzato il percorso più breve tra le destinazioni. Network Access Analyzer può aiutarti a farlo.

Risorse

Documenti correlati:

- Registrazione delle query DNS pubbliche
- Che cos'è IPAM?
- What is Reachability Analyzer?
- What is Network Access Analyzer?
- Parametri di CloudWatch per i VPC
- Ottimizzazione delle prestazioni e riduzione dei costi per l'analisi della rete con log di flusso VPC in formato Apache Parquet
- Monitoring your global and core networks with Amazon CloudWatch metrics
- Continuously monitor network traffic and resources

Video correlati:

- AWS re:Invent 2023 – A developer's guide to cloud networking
- AWS re:Invent 2023 – Ready for what’s next? Designing networks for growth and flexibility
- AWS re:Invent 2023 – Advanced VPC designs and new capabilities

- AWS re:Invent 2022 – Dive deep on AWS networking infrastructure
- AWS re:Invent 2020 – Networking best practices and tips with the AWS Well-Architected Framework
- AWS re:Invent 2020 – Monitoring and troubleshooting network traffic

Esempi correlati:

- AWS Networking Workshops
- AWS Network Monitoring
- Observing and diagnosing your network on AWS
- Finding and addressing network misconfigurations on AWS

Processo e cultura

PERF 5. In che modo le pratiche e la cultura dell'organizzazione contribuiscono all'efficienza delle prestazioni nel carico di lavoro?

Durante la fase di progettazione dei carichi di lavoro, esistono principi e pratiche che è possibile adottare per gestire al meglio carichi di lavoro cloud efficienti e ad alte prestazioni. Per adottare una cultura che promuova l'efficienza delle prestazioni dei carichi di lavoro cloud, prendi in considerazione questi principi e pratiche fondamentali:

Best practice

- PERF05-BP01 Individuazione degli indicatori chiave di prestazioni (KPI) per misurare l'integrità e le prestazioni del carico di lavoro
- PERF05-BP02 Uso di soluzioni di monitoraggio per comprendere le aree in cui le prestazioni sono più critiche
- PERF05-BP03 Definizione di un processo per migliorare le prestazioni del carico di lavoro
- PERF05-BP04 Esecuzione del test del carico di lavoro
- PERF05-BP05 Uso dell'automazione per risolvere in modo proattivo i problemi relativi alle prestazioni
- PERF05-BP06 Aggiornamento continuo del carico di lavoro e dei servizi
- PERF05-BP07 Analisi dei parametri a intervalli regolari

PERF05-BP01 Individuazione degli indicatori chiave di prestazioni (KPI) per misurare l'integrità e le prestazioni del carico di lavoro

Individua gli indicatori chiave di prestazione (KPI) per misurare le prestazioni del carico di lavoro. I KPI consentono di misurare l'integrità e le prestazioni di un carico di lavoro correlato a un obiettivo aziendale.

Anti-pattern comuni:

- Monitori i parametri a livello di sistema solo per avere una visione del carico di lavoro e non valuti gli impatti aziendali di tali parametri.
- Ritieni che i KPI siano già in fase di pubblicazione e condivisi come dati parametrici standard.
- Non definisci un KPI quantitativo e misurabile.
- Non esegui l'allineamento dei KPI a obiettivi o strategie aziendali.

Vantaggi dell'adozione di questa best practice: l'individuazione di KPI specifici che rappresentino l'integrità e le prestazioni del carico di lavoro aiuta ad allineare i team alle priorità e a definire risultati aziendali ottimali. La condivisione di tali parametri con tutti i reparti fornisce visibilità e allineamento su soglie, aspettative e impatto aziendale.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Gli indicatori chiave di prestazione consentono ai team aziendali e di ingegneri di allinearsi sulla misurazione degli obiettivi e delle strategie e sul modo in cui questi fattori si combinano per produrre risultati aziendali. Ad esempio, il carico di lavoro di un sito Web può utilizzare il tempo di caricamento della pagina come indicazione delle prestazioni complessive. Questa metrica sarebbe uno dei molteplici punti dati che misurano l'esperienza dell'utente. Oltre a identificare le soglie di tempo di caricamento della pagina, è necessario documentare il risultato atteso o il rischio aziendale se le prestazioni ideali non vengono rispettate. Un lungo tempo di caricamento della pagina si ripercuote direttamente sugli utenti finali, diminuisce la loro esperienza d'uso e può portare a una perdita di clienti. Quando definisci le soglie degli indicatori chiave di prestazione, devi combinare sia i benchmark di settore sia le aspettative degli utenti finali. Ad esempio, se l'attuale benchmark del settore prevede il caricamento di una pagina Web entro un periodo di tempo di due secondi, ma gli utenti finali si aspettano che la pagina Web venga caricata entro un periodo di tempo di un secondo, allora devi prendere in considerazione entrambi i dati al momento di stabilire l'indicatore chiave di prestazione (KPI).

Il team deve valutare i KPI del carico di lavoro utilizzando dati granulari in tempo reale e dati storici di riferimento e creare pannelli di controllo che eseguano calcoli metrici sui dati KPI per ricavare informazioni operative e di utilizzo. I KPI devono essere documentati e includere le soglie che supportano gli obiettivi e le strategie aziendali e che sono mappati sui parametri da monitorare. Gli indicatori chiave di prestazione devono essere riesaminati quando cambiano gli obiettivi aziendali, le strategie o i requisiti degli utenti finali.

Passaggi dell'implementazione

- Individua le parti interessate: identifica e documenta le principali parti interessate aziendali, compresi i team di sviluppo e operativi.
- Definisci gli obiettivi: collabora con queste parti interessate per definire e documentare gli obiettivi del carico di lavoro. Considera gli aspetti critici relativi alle prestazioni dei carichi di lavoro, come il throughput, i tempi di risposta e i costi, nonché gli obiettivi aziendali, come la soddisfazione degli utenti.
- Esamina le best practice del settore: esamina le best practice del settore per individuare i KPI pertinenti in linea con gli obiettivi del carico di lavoro.
- Individua le metriche: identifica le metriche che sono allineate agli obiettivi del carico di lavoro e che possono aiutarti a misurare le prestazioni e gli obiettivi aziendali. Stabilisci i KPI in base a queste metriche, ad esempio le misurazioni del tempo medio di risposta o del numero di utenti simultanei.
- Definisci e documenta i KPI: utilizza le best practice del settore e gli obiettivi del carico di lavoro per stabilire i valori dei KPI del carico di lavoro. Utilizza queste informazioni per impostare soglie dei KPI per livello di gravità o allarme. Individua e documenta il rischio e l'impatto se il KPI non viene raggiunto.
- Implementa il monitoraggio: utilizza gli strumenti di monitoraggio come Amazon CloudWatch o AWS Config per raccogliere le metriche e misurare i KPI.
- Comunica visivamente i KPI: utilizza gli strumenti della dashboard, come Amazon QuickSight, per visualizzare e comunicare i KPI alle parti interessate.
- Analizza e ottimizza: esamina e analizza regolarmente i KPI per identificare le aree del carico di lavoro che devono essere migliorate. Collabora con le parti interessate per implementare i miglioramenti.
- Riesamina e perfeziona: revisiona regolarmente le metriche e i KPI per valutarne l'efficacia, soprattutto quando cambiano gli obiettivi aziendali o le prestazioni del carico di lavoro.

Risorse

Documenti correlati:

- Documentazione CloudWatch
- Monitoraggio, registrazione di log e prestazioni - AWS Partner
- AWS observability tools
- The Importance of Key Performance Indicators (KPIs) for Large-Scale Cloud Migrations
- How to track your cost optimization KPIs with the KPI Dashboard
- DocumentazioneX-Ray
- Using Amazon CloudWatch dashboards
- Amazon QuickSight KPIs

Video correlati:

- AWS re:Invent 2023 - Optimize cost and performance and track progress toward mitigation
- AWS re:Invent 2023 - Manage resource lifecycle events at scale with AWS Health
- AWS re:Invent 2023 - Performance & efficiency at Pinterest: Optimizing the latest instances
- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results
- AWS re:Invent 2023 - Building an effective observability strategy
- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS
- AWS re:Invent 2023 - Scaling on AWS for the first 10 million users
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance
- Creating an Effective Metrics Strategy for Your Business | AWS Events

Esempi correlati:

- Creating a dashboard with Amazon QuickSight

PERF05-BP02 Uso di soluzioni di monitoraggio per comprendere le aree in cui le prestazioni sono più critiche

Comprendi e identifica le aree in cui l'aumento delle prestazioni del carico di lavoro determinerà un impatto positivo sull'efficienza o sull'esperienza del cliente. Ad esempio, un sito web che ha

una grande quantità di interazione con i clienti può trarre vantaggio dall'utilizzo dei servizi edge per spostare la distribuzione di contenuti più vicino ai clienti.

Anti-pattern comuni:

- Ritieni che i parametri di calcolo standard, ad esempio l'utilizzo della CPU o il carico della memoria, siano sufficienti per rilevare problemi di prestazioni.
- Utilizzi solo i parametri predefiniti registrati dal software di monitoraggio selezionato.
- Rivedi i parametri solo quando c'è un problema.

Vantaggi dell'adozione di questa best practice: la comprensione delle aree critiche delle prestazioni aiuta i proprietari dei carichi di lavoro a monitorare i KPI e a dare priorità ai miglioramenti ad alto impatto.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Configura il tracciamento end-to-end per identificare gli schemi di traffico, la latenza e le aree con prestazioni critiche. Monitora gli schemi di accesso ai dati per query lente o dati scarsamente frammentati e partizionati. Identifica le aree vincolate del carico di lavoro utilizzando il test o il monitoraggio del carico.

aumenta l'efficienza delle prestazioni comprendendo l'architettura, gli schemi di traffico e gli schemi di accesso ai dati e identifica la latenza e i tempi di elaborazione. Identifica i potenziali colli di bottiglia che potrebbero influire sull'esperienza del cliente man mano che il carico di lavoro aumenta. Dopo aver identificato queste aree, individua quale soluzione puoi implementare per evitare tali problemi di prestazioni.

Passaggi dell'implementazione

- Configura il monitoraggio end-to-end per acquisire tutti i componenti e i parametri del carico di lavoro. Ecco alcuni esempi di soluzioni di monitoraggio su AWS.

Service	Where to use
Amazon CloudWatch Real-User Monitoring (RUM)	To capture application performance metrics from real user client-side and frontend sessions.

Service	Where to use
AWS X-Ray	To trace traffic through the application layers and identify latency between components and dependencies. Use X-Ray service maps to see relationships and latency between workload components.
Amazon Relational Database Service Performance Insights	To view database performance metrics and identify performance improvements.
Monitoraggio dei parametri del sistema operativo con il monitoraggio avanzato - Amazon RDS	To view database OS performance metrics.
Amazon DevOps Guru	To detect abnormal operating patterns so you can identify operational issues before they impact your customers.

- Esegui i test per generare parametri, identificare schemi di traffico, colli di bottiglia e aree con prestazioni critiche. Ecco alcuni esempi di come eseguire i test:
  - Configura i canary Synthetics di CloudWatch per simulare le attività degli utenti basate sul browser in modo programmatico utilizzando espressioni di valutazione o processi CRON di Linux per generare parametri coerenti nel tempo.
  - Utilizza la soluzione per il test di carico distribuito AWS per generare picchi di traffico o testare il carico di lavoro al tasso di crescita previsto.
- Valuta i parametri e i dati di telemetria per identificare le aree critiche delle prestazioni. Esamina queste aree con il tuo team per determinare il monitoraggio e le soluzioni per evitare i colli di bottiglia.
- Sperimenta i miglioramenti delle prestazioni e valuta tali modifiche con i dati. Ad esempio, puoi usare CloudWatch Evidently per testare i nuovi miglioramenti e l'impatto sulle prestazioni del tuo carico di lavoro.

Risorse

Documenti correlati:

- What's new in AWS Observability at re:Invent 2023
- Amazon Builders' Library
- DocumentazioneX-Ray
- Amazon CloudWatch RUM
- Amazon DevOps Guru

Video correlati:

- AWS re:Invent 2023 - [LAUNCH] Application monitoring for modern workloads
- AWS re:Invent 2023 - Implementing application observability
- AWS re:Invent 2023 - Building an effective observability strategy
- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS
- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results
- AWS re:Invent 2022 - The Amazon Builders’ Library: 25 years of Amazon operational excellence
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance
- Visual Monitoring of Applications with Amazon CloudWatch Synthetics

Esempi correlati:

- Measure page load time with Amazon CloudWatch Synthetics
- Amazon CloudWatch RUM Web Client
- X-Ray SDK for Python
- Distributed Load Testing on AWS

PERF05-BP03 Definizione di un processo per migliorare le prestazioni del carico di lavoro

Definisci un processo per valutare i nuovi servizi, i modelli di progettazione, i tipi di risorse e le configurazioni man mano che diventano disponibili. Ad esempio, esegui test delle prestazioni esistenti sulle nuove offerte di istanze per determinare il loro potenziale per migliorare il carico di lavoro.

Anti-pattern comuni:

- Ritieni che l'architettura corrente diventi statica e non venga aggiornata nel corso del tempo.
- Introduci modifiche all'architettura nel tempo senza dei parametri che le giustifichino.

Vantaggi dell'adozione di questa best practice: definendo il processo per apportare modifiche all'architettura, puoi utilizzare i dati raccolti per influenzare la progettazione del carico di lavoro nel tempo.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Le prestazioni del carico di lavoro presentano alcuni vincoli principali. Documentali, in modo da sapere quali tipi di innovazione potrebbero migliorare le prestazioni del carico di lavoro. Utilizza queste informazioni quando vieni a conoscenza di nuovi servizi e tecnologie, man mano che si rendono disponibili, in modo da identificare le soluzioni per ovviare ai vincoli o ai colli di bottiglia.

Determina i principali vincoli riguardanti le prestazioni del carico di lavoro. Documenta i vincoli prestazionali del carico di lavoro in modo da sapere quali tipi di innovazione potrebbero migliorare le prestazioni del carico di lavoro.

Passaggi dell'implementazione

- Individua i KPI: determina i KPI delle prestazioni del carico di lavoro come indicato in PERF05-BP01 Individuazione degli indicatori chiave di prestazioni (KPI) per misurare l'integrità e le prestazioni del carico di lavoro per definire la baseline del carico di lavoro.
- Implementa il monitoraggio: utilizza gli strumenti di osservabilità AWS per raccogliere le metriche sulle prestazioni e misurare i KPI.
- Esegui l'analisi: conduci un'analisi approfondita per individuare le aree del carico di lavoro, ad esempio la configurazione e il codice applicativo, con prestazioni insufficienti, come indicato in PERF05-BP02 Uso di soluzioni di monitoraggio per comprendere le aree in cui le prestazioni sono più critiche. Usa i tuoi strumenti di analisi e prestazioni per individuare la strategia di miglioramento delle prestazioni.
- Convalida i miglioramenti: utilizza ambienti sandbox o di pre-produzione per convalidare l'efficacia delle strategie di miglioramento.
- Apporta le modifiche: implementa le modifiche in produzione e monitora continuamente le prestazioni del carico di lavoro. Documenta i miglioramenti e comunica i risultati alle parti interessate.
- Riesamina e perfeziona: revisiona regolarmente il processo di miglioramento delle prestazioni per identificare le aree da potenziare.

Risorse

Documenti correlati:

- Blog AWS
- Novità di AWS
- AWS Skill Builder

Video correlati:

- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2023 - Optimize cost and performance and track progress toward mitigation
- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results
- AWS re:Invent 2022 - Optimize your AWS workloads with best-practice guidance

Esempi correlati:

- AWS Github

PERF05-BP04 Esecuzione del test del carico di lavoro

Esegui il test del carico di lavoro per verificare che sia in grado di gestire la produzione e individuare eventuali colli di bottiglia nelle prestazioni.

Anti-pattern comuni:

- Vengono testate le singole parti del carico di lavoro, ma non l'intero carico di lavoro.
- Il test di carico viene eseguito su un'infrastruttura diversa dall'ambiente di produzione.
- Esegui i test di carico solo per il carico previsto e non oltre, per prevedere dove si potrebbero riscontrare problemi futuri.
- Esegui il test di carico senza consultare la policy di test di Amazon EC2 e presentare un modulo di invio di eventi simulati. Ciò comporta la mancata esecuzione del test, poiché sembra un evento di negazione del servizio.

Vantaggi dell'adozione di questa best practice: misurando le prestazioni con un test di carico puoi osservare dove avrà luogo l'impatto dell'aumento del carico. In questo modo puoi anticipare le modifiche necessarie prima che influiscano sul carico di lavoro.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

Il test di carico nel cloud è un processo per misurare le prestazioni del carico di lavoro in condizioni realistiche e con il carico degli utenti previsto. Questo processo prevede il provisioning di un ambiente cloud simile a quello di produzione, l'utilizzo di strumenti di test di carico per generare il carico e l'analisi dei parametri per valutare la capacità del carico di lavoro di gestire un carico realistico. Occorre eseguire i test di carico tramite versioni sintetiche o purificate dei dati di produzione (rimuovendo le informazioni sensibili o che permettono l'identificazione degli utenti). Esegui automaticamente test di carico come parte della pipeline di distribuzione e confronta i risultati con KPI e soglie predefiniti. Questo processo ti consente di ottenere le prestazioni richieste.

Passaggi dell'implementazione

- Definisci gli obiettivi del test: individua gli aspetti prestazionali del carico di lavoro che desideri valutare, come il throughput e il tempo di risposta.
- Seleziona uno strumento di test: scegli e configura lo strumento di test più adatto al tuo carico di lavoro.
- Configura l'ambiente: configura l'ambiente di test in base all'ambiente di produzione. Puoi utilizzare i servizi AWS per eseguire ambienti in ambito di produzione e sottoporre l'architettura a test.
- Implementa il monitoraggio: utilizza gli strumenti di monitoraggio, ad esempio Amazon CloudWatch, per raccogliere le metriche delle risorse della tua architettura. Puoi anche raccogliere e pubblicare metriche personalizzate.
- Definisci gli scenari: stabilisci gli scenari e i parametri del test di carico, come la durata del test e il numero di utenti.
- Conduci il test di carico: esegui gli scenari di test su larga scala. Sfrutta i vantaggi offerti dal Cloud AWS per testare il carico di lavoro e scoprire dove la scalabilità non è possibile o se non è lineare. Ad esempio, usa le istanze Spot per generare carichi a costi ridotti e rilevare i colli di bottiglia prima che si verifichino in produzione.
- Analizza i risultati del test: analizza i risultati per individuare i colli di bottiglia delle prestazioni e le aree di miglioramento.

- Documenta e condividi gli esiti: crea i documenti per comunicare i risultati e le raccomandazioni. Condividi queste informazioni con le parti interessate per aiutarle a prendere decisioni informate sulle strategie di ottimizzazione delle prestazioni.
- Itera in modo continuo: i test di carico devono essere eseguiti a cadenza regolare, soprattutto dopo una modifica o un aggiornamento del sistema.

Risorse

Documenti correlati:

- Amazon CloudWatch RUM
- Amazon CloudWatch Synthetics
- Distributed Load Testing on AWS

Video correlati:

- AWS Summit ANZ 2023: Accelerate with confidence through AWS Distributed Load Testing
- AWS re:Invent 2022 - Scaling on AWS for your first 10 million users
- Solving with AWS Solutions: Distributed Load Testing

- AWS re:Invent 2021 - Ottimizzare le applicazioni con gli approfondimenti degli utenti finali con Amazon CloudWatch RUM
- Demo di Amazon CloudWatch Synthetics

Esempi correlati:

- Distributed Load Testing on AWS

PERF05-BP05 Uso dell'automazione per risolvere in modo proattivo i problemi relativi alle prestazioni

Utilizza indicatori chiave di prestazioni (KPI), in combinazione con sistemi di monitoraggio e allarmi, per risolvere in modo proattivo i problemi correlati alle prestazioni.

Anti-pattern comuni:

- Consenti solo al personale operativo di apportare modifiche operative al carico di lavoro.

- Lasci che tutti gli allarmi giungano direttamente al team operativo senza alcuna correzione proattiva.

Vantaggi dell'adozione di questa best practice: la correzione proattiva delle azioni di allarme consente al team di supporto di concentrarsi sugli elementi che non sono attivabili automaticamente. In questo modo, il personale operativo non viene sovraccaricato da tutti gli allarmi e si concentra, invece, solo sugli allarmi critici.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

Laddove possibile, utilizza gli allarmi per attivare operazioni automatizzate per risolvere i problemi. Se non è possibile rispondere in modo automatizzato, inoltra l'allarme a coloro che possono intervenire. Ad esempio, puoi implementare un sistema in grado di prevedere i valori attesi per gli indicatori chiave di prestazioni (KPI) e di inviare allarmi qualora essi oltrepassino determinate soglie, oppure uno strumento che arresta o esegue automaticamente il rollback delle implementazioni nel caso in cui i valori dei KPI si discostino dai valori attesi.

Implementa processi che forniscono visibilità sulle prestazioni durante l'esecuzione del carico di lavoro. Crea pannelli di controllo del monitoraggio e stabilisci norme di riferimento per le aspettative riguardanti le prestazioni, per determinare se il carico di lavoro ha prestazioni ottimali.

Passaggi dell'implementazione

- Individua il flusso di lavoro della risoluzione: identifica e comprendi il problema delle prestazioni che può essere risolto automaticamente. Utilizza soluzioni di monitoraggio AWS come Amazon CloudWatch o AWS X-Ray per comprendere meglio la causa principale del problema.
- Definisci il processo di automazione: crea un processo di risoluzione dettagliato che possa essere utilizzato per risolvere automaticamente il problema.
- Configura l'evento di avvio: definisci l'evento che avvia automaticamente il processo di risoluzione. Ad esempio, è possibile definire un trigger per riavviare automaticamente un'istanza quando raggiunge una determinata soglia di utilizzo della CPU.
- Automatizza la risoluzione: utilizza i servizi e le tecnologie AWS per automatizzare il processo di risoluzione. Ad esempio, l'automazione AWS Systems Manager fornisce un modo sicuro e dimensionabile per automatizzare il processo di risoluzione. Assicurati di utilizzare la logica di risoluzione automatica per annullare le modifiche se non risolvono correttamente il problema.

- Esegui il test del flusso di lavoro: esegui il test del processo di risoluzione automatizzato in un ambiente di pre-produzione.
- Implementa il flusso di lavoro: implementa la risoluzione automatizzata nell'ambiente di produzione.
- Sviluppa un playbook: crea e documenta un playbook che delinei i passaggi per il piano di risoluzione, inclusi gli eventi di avvio, la logica di risoluzione e le azioni intraprese. Assicurati di fornire la giusta preparazione alle parti interessate per aiutarle a rispondere efficacemente agli eventi di risoluzione automatizzati.
- Rivedi e perfeziona: valuta regolarmente l'efficacia del flusso di lavoro di risoluzione automatizzato. Modifica gli eventi di avvio e la logica di risoluzione, se necessario.

Risorse

Documenti correlati:

- Documentazione CloudWatch
- Monitoraggio, registrazione di log e prestazioni - Partner AWS Partner Network
- DocumentazioneX-Ray
- Using Alarms and Alarm Actions in CloudWatch
- Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services
- Automate your Amazon Redshift performance tuning with automatic table optimization

Video correlati:

- AWS re:Invent 2023 - Strategies for automated scaling, remediation, and smart self-healing
- AWS re:Invent 2023 - [LAUNCH] Application monitoring for modern workloads
- AWS re:Invent 2023 - Implementing application observability
- AWS re:Invent 2021 - Intelligently automating cloud operations
- AWS re:Invent 2022 - Setting up controls at scale in your AWS environment
- AWS re:Invent 2022 - Automating patch management and compliance using AWS
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance
- AWS re:Invent 2023 - Take a load off: Diagnose & resolve performance issues with Amazon RDS
- AWS re:Invent 2021 -{New Launch} Automatically detect and resolve issues with Amazon DevOps Guru

- AWS re:Invent 2023 - Centralize your operations

Esempi correlati:

- CloudWatch Logs Customize Alarms

PERF05-BP06 Aggiornamento continuo del carico di lavoro e dei servizi

Rimani aggiornato sui nuovi servizi cloud per adottare funzionalità efficienti, rimuovere i problemi e migliorare l'efficienza complessiva delle prestazioni del tuo carico di lavoro.

Anti-pattern comuni:

- Ritieni che l'architettura corrente diventi statica e non venga aggiornata nel corso del tempo.
- Non disponi di sistemi né esegui regolarmente una valutazione per la compatibilità di software e pacchetti aggiornati con il carico di lavoro.

Vantaggi dell'adozione di questa best practice: stabilendo un processo per rimanere aggiornati su nuovi servizi e offerte, è possibile adottare nuove capacità e funzionalità, risolvere problemi e migliorare le prestazioni dei carichi di lavoro.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

Valuta i modi per migliorare le prestazioni man mano che nuovi servizi, modelli di progettazione e funzionalità di prodotti diventano disponibili. Determina come possono migliorare le prestazioni o aumentare l'efficienza del carico di lavoro tramite una valutazione, una discussione interna o un'analisi esterna. Definisci un processo per valutare gli aggiornamenti, le nuove funzioni e i servizi rilevanti per il tuo carico di lavoro. Ad esempio, crea un proof of concept che utilizza le nuove tecnologie o consultati con un gruppo interno. Quando provi nuove idee o servizi, esegui i test delle prestazioni per misurare l'impatto del carico di lavoro sulle prestazioni.

Passaggi dell'implementazione

- Esegui l'inventario del carico di lavoro: redigi l'inventario del software e dell'architettura del carico di lavoro e identifica i componenti che richiedono un aggiornamento.
- Individua le origini di aggiornamento: identifica le origini di notizie e aggiornamenti relative ai componenti del carico di lavoro. Ad esempio, puoi iscriverti al blog Novità di AWS per scoprire i

prodotti che corrispondono al tuo componente del carico di lavoro. Puoi iscriverti al feed RSS o gestire le tue sottoscrizioni e-mail.

- Definisci una pianificazione degli aggiornamenti: stabilisci la pianificazione per valutare nuovi servizi e funzionalità per il carico di lavoro.
  - Puoi usare AWS Systems Manager Inventory per raccogliere i metadati relativi a sistema operativo (SO), applicazioni e istanze dalle istanze Amazon EC2 per avere una panoramica immediata su quali istanze stanno eseguendo il software e le configurazioni richieste dalle policy software e quali istanze devono essere aggiornate.
- Valuta il nuovo aggiornamento: individua le modalità di aggiornamento dei componenti del carico di lavoro. Sfrutta l'agilità del cloud per testare in modo semplice e rapido il modo in cui le nuove funzionalità possono migliorare il carico di lavoro per ottenere efficienza delle prestazioni.
- Usa l'automazione: utilizza l'automazione del processo di aggiornamento per ridurre il livello di impegno per implementare le nuove funzionalità e limitare gli errori causati dai processi manuali.
  - Puoi usare CI/CD per aggiornare automaticamente le AMI, le immagini di container e altri artefatti relativi alla tua applicazione cloud.
  - Puoi usare strumenti come AWS Systems Manager Patch Manager per automatizzare il processo degli aggiornamenti di sistema e pianificare le attività tramite Finestre di manutenzione AWS Systems Manager.
- Documenta il processo: crea i documenti per il processo di valutazione degli aggiornamenti e dei nuovi servizi. Fornisci ai proprietari il tempo e lo spazio necessari per ricercare, testare, sperimentare e convalidare aggiornamenti e nuovi servizi. Fai riferimento ai requisiti aziendali e ai KPI documentati per stabilire la priorità dell'aggiornamento che avrà un impatto positivo sull'azienda.

Risorse

Documenti correlati:

- Blog AWS
- Novità di AWS
- Implementing up-to-date images with automated EC2 Image Builder pipelines

Video correlati:

- AWS re:Inforce 2022 - Automating patch management and compliance using AWS

- All Things Patch: AWS Systems Manager | AWS Events

Esempi correlati:

- Inventory and Patch Management
- One Observability Workshop

PERF05-BP07 Analisi dei parametri a intervalli regolari

Come manutenzione ordinaria o in risposta a eventi o incidenti, esamina quali parametri vengono raccolti. Stabilisci quali di questi parametri sono fondamentali per risolvere i problemi e quali altri parametri aggiuntivi, se monitorati, possono contribuire a identificare, affrontare o prevenire i problemi.

Anti-pattern comuni:

- Lasci che i parametri rimangano in uno stato di allarme per un lungo periodo di tempo.
- Crei allarmi che non sono utilizzabili da un sistema di automazione.

Vantaggi dell'adozione di questa best practice: esamina continuamente i parametri raccolti per verificare che individuano, risolvano o prevengano correttamente i problemi. I parametri possono anche diventare obsoleti se lasciati in uno stato di allarme per un lungo periodo di tempo.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Migliora continuamente la raccolta e il monitoraggio dei parametri. Nell'ambito della risposta a incidenti ed eventi, valuta quali parametri sono stati utili per affrontare il problema e quali sarebbero stati utili ma non sono attualmente misurati. Questo metodo ti aiuterà a migliorare la qualità dei parametri raccolti, in modo da prevenire o risolvere più rapidamente gli incidenti futuri.

Nell'ambito della risposta a incidenti ed eventi, valuta quali parametri sono stati utili per affrontare il problema e quali sarebbero stati utili ma non sono attualmente misurati. Queste considerazioni ti aiuteranno a migliorare la qualità dei parametri raccolti, per prevenire o risolvere più rapidamente gli incidenti futuri.

Passaggi dell'implementazione

- Definisci le metriche: individua le metriche prestazionali critiche da monitorare affinché siano allineate all'obiettivo del carico di lavoro, ad esempio il tempo di risposta e l'utilizzo delle risorse.
- Stabilisci le baseline: imposta una baseline e il valore desiderabile per ogni metrica. La baseline deve fornire i punti di riferimento per identificare deviazioni o anomalie.
- Imposta una cadenza: stabilisci una cadenza, ad esempio settimanale o mensile, per la revisione delle metriche critiche.
- Individua i problemi relativi alle prestazioni: durante ogni revisione, valuta le tendenze e le deviazioni dai valori della baseline. Cerca eventuali rallentamenti o anomalie nelle prestazioni. Per i problemi identificati, esegui un'analisi approfondita delle cause principali per comprendere il motivo più importante alla base del problema.
- Individua le azioni correttive: utilizza l'analisi per individuare le azioni correttive, come l'ottimizzazione dei parametri, la correzione di bug e il dimensionamento delle risorse.
- Documenta gli esiti: crea i documenti per comunicare gli esiti, inclusi i problemi identificati, le cause principali e le azioni correttive.
- Itera e migliora: valuta e perfeziona continuamente il processo di revisione delle metriche. Usa le indicazioni apprese dalla revisione precedente per migliorare il processo nel tempo.

Risorse

Documenti correlati:

- Documentazione CloudWatch
- Collect metrics and logs from Amazon EC2 Instances and on-premises servers with the CloudWatch Agent
- Query your metrics with CloudWatch Metrics Insights
- Monitoraggio, registrazione di log e prestazioni - Partner AWS Partner Network
- DocumentazioneX-Ray

Video correlati:

- AWS re:Invent 2022 - Setting up controls at scale in your AWS environment
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance

- AWS re:Invent 2023 - Building an effective observability strategy
- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS
- AWS re:Invent 2023 - Take a load off: Diagnose & resolve performance issues with Amazon RDS

Esempi correlati:

- Creating a dashboard with Amazon QuickSight
- CloudWatch Dashboards

Ottimizzazione dei costi

Il principio dell'ottimizzazione dei costi include la possibilità di eseguire sistemi per offrire valore aggiunto al prezzo più basso. È possibile trovare linee guida prescrittive sull'implementazione nel Whitepaper sul principio dell'ottimizzazione dei costi.

Aree delle best practice

- Implementazione della gestione finanziaria del cloud
- Comprensione delle spese e dell'utilizzo
- Risorse a costi contenuti
- Gestione delle risorse di domanda e offerta
- Ottimizzazione nel tempo

Implementazione della gestione finanziaria del cloud

Domanda

- COST 1. Come implementi la gestione finanziaria nel cloud?

COST 1. Come implementi la gestione finanziaria nel cloud?

L'implementazione della gestione finanziaria del cloud aiuta le organizzazioni a conseguire un valore aggiunto e il successo finanziario ottimizzando i costi e l'utilizzo e dimensionando le risorse in AWS.

Best practice

- COST01-BP01 Stabilire la responsabilità dell'ottimizzazione dei costi

- COST01-BP02 Definizione di una partnership tra team finanziari e tecnologici
- COST01-BP03 Definizione di budget e previsioni per il cloud
- COST01-BP04 Implementazione della consapevolezza dei costi nei processi dell'organizzazione
- COST01-BP05 Invio di report e notifiche sull'ottimizzazione dei costi
- COST01-BP06 Monitoraggio proattivo dei costi
- COST01-BP07 Mantenimento dell'aggiornamento sulle nuove versioni dei servizi
- COST01-BP08 Creazione di una cultura consapevole dei costi
- COST01-BP09 Quantificare il valore aggiunto realizzato attraverso l'ottimizzazione dei costi

COST01-BP01 Stabilire la responsabilità dell'ottimizzazione dei costi

Crea un team (Cloud Business Office, Cloud Center of Excellence, o FinOps) responsabile di stabilire e mantenere la consapevolezza dei costi in tutta l'organizzazione. Il responsabile dell'ottimizzazione dei costi può essere un individuo o un team (sono necessarie persone provenienti da team finanziari, tecnologici e aziendali) che ha una comprensione dell'intera organizzazione e degli aspetti finanziari legati al cloud.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Questa è un'introduzione al processo di formazione di un team Ufficio aziendale per il cloud (Cloud Business Office, CBO) o Centro di eccellenza del cloud (Cloud Center of Excellence, CCoE) responsabile di stabilire e mantenere una cultura basata sulla consapevolezza dei costi legati al cloud computing. Può trattarsi di una figura professionale già in organico, di un team all'interno della tua organizzazione o di un nuovo team di stakeholder chiave dei settori finanza, tecnologia e organizzazione provenienti da tutta l'azienda.

La funzione (individuo o team) stabilisce le priorità e dedica la parte prevista del proprio tempo alle attività di gestione e ottimizzazione dei costi. In un'organizzazione di dimensioni ridotte, la quantità di tempo dedicata dalla funzione potrebbe essere inferiore rispetto a quella dedicata da una funzione a tempo pieno in un'azienda di dimensioni maggiori.

Questa funzione (individuo o team) stabilisce le priorità e dedica la parte prevista del proprio tempo alle attività di gestione e ottimizzazione dei costi. Per una piccola organizzazione, la funzione potrebbe dedicare una percentuale di tempo inferiore alle attività di gestione e ottimizzazione dei costi rispetto a una funzione a tempo pieno per un'azienda più grande.

La funzione richiede un approccio multidisciplinare, con capacità di gestione dei progetti, data science, analisi finanziaria e sviluppo di software o infrastruttura. Può migliorare l'efficienza del carico di lavoro eseguendo ottimizzazioni dei costi all'interno di tre diversi tipi di responsabilità:

- Team centralizzati: attraverso team designati come il team FinOps, il team Cloud Financial Management (CFM), l'Ufficio aziendale per il cloud (Cloud Business Office, CBO) o il Centro di eccellenza del cloud (Cloud Center of Excellence, CCoE), i clienti possono progettare e implementare meccanismi di governance e promuovere le best practice a livello aziendale.
- Team decentralizzati: Influenzando i team tecnologici per ottimizzare i costi.
- Team ibridi: Una combinazione di team centralizzati e decentralizzati può collaborare fattivamente per eseguire l'ottimizzazione dei costi.

La funzione può essere valutata in base alla sua capacità di eseguire e conseguire risultati rispetto agli obiettivi di ottimizzazione dei costi (ad esempio in base a parametri di efficienza dei carichi di lavoro).

Un fattore chiave per il successo di questa funzione è la disponibilità di sponsorizzazione da parte del management. Lo sponsor deve essere un sostenitore del consumo efficiente del cloud e fornire alla funzione un supporto in caso di escalation, per garantire che le attività di ottimizzazione dei costi vengano trattate con il livello di priorità definito dall'organizzazione. In caso contrario, le linee guida possono essere ignorate e non verrà data priorità alle opportunità di riduzione dei costi. Insieme, lo sponsor (cioè la figura o le figure di garanzia all'interno del management) e il team dell'organizzazione possono aiutare a utilizzare il cloud in modo efficiente e generare valore aziendale.

Se hai un piano di supporto Business, Enterprise-On-Ramp o Enterprise e hai bisogno di aiuto per creare questo team o questa funzione, contatta i tuoi esperti di Cloud Financial Management (CFM) tramite il team del tuo account.

Passaggi dell'implementazione

- Definizione dei membri chiave: tutte le parti rilevanti della tua organizzazione devono contribuire ed essere interessate alla gestione dei costi. I team più comuni all'interno delle organizzazioni includono in genere: team finanziari, responsabili di applicazioni o prodotti, team di gestione e team tecnici (DevOps). Alcuni soggetti sono impegnati a tempo pieno (ad esempio quelli di tipo finanziario o tecnico), mentre altri sono coinvolti periodicamente secondo necessità. I singoli o i team che svolgono mansioni di gestione finanziaria del cloud devono disporre del seguente set di competenze:

  - Sviluppo software: nel caso in cui vengano creati script e automazione.
  - Progettazione dell'infrastruttura: per implementare script, automatizzare processi e comprendere in che modo viene effettuato il provisioning di risorse e servizi.
  - Acume operativo: gestione finanziaria del cloud significa una presenza efficiente nel cloud mediante la misurazione, il monitoraggio, la modifica, la pianificazione e il dimensionamento dell'utilizzo efficiente del cloud stesso.
- Definizione di obiettivi e parametri: La funzione deve fornire valore all'organizzazione in modi diversi. Questi obiettivi sono definiti e si evolvono continuamente con l'evolversi dell'organizzazione. Tra le attività più comuni figurano la creazione e l'esecuzione di programmi di formazione sull'ottimizzazione dei costi in tutta l'organizzazione, lo sviluppo di standard a livello aziendale, come monitoraggio ed elaborazione di report per l'ottimizzazione dei costi, e la definizione degli obiettivi di ottimizzazione dei carichi di lavoro. Inoltre, è necessario comunicare regolarmente all'organizzazione la relativa capacità di ottimizzazione dei costi.

  È possibile definire gli indicatori chiave delle prestazioni (KPI) basati sui valori o sui costi. Quando vengono definiti i KPI, è possibile calcolare il costo previsto in termini di efficienza e risultati aziendali previsti. I KPI basati sui valori associano le metriche relative a costi e utilizzo ai fattori legati al valore aziendale e ciò aiuta a razionalizzare le modifiche ai livelli di spesa per AWS. Il primo passo per derivare i KPI basati sui valori è collaborare tra organizzazioni al fine di scegliere e concordare un set standard di KPI.
- Definizione di una regolare cadenza: il gruppo (team finanziario, tecnologico e aziendale) deve riunirsi regolarmente per rivedere le metriche e gli obiettivi. Una periodicità tipica implica la revisione dello stato dell'organizzazione, la revisione dei programmi attualmente in esecuzione e la revisione dei parametri finanziari e di ottimizzazione generali. Quindi, per i carichi di lavoro chiave, è opportuno elaborare report più dettagliati.

  Durante queste riunioni periodiche è possibile analizzare l'efficienza (costo) dei carichi di lavoro e il risultato aziendale. Ad esempio, un incremento del 20% dei costi di un carico di lavoro potrebbe essere determinato dall'aumento dell'utilizzo da parte dei clienti. In questo caso, l'incremento del 20% dei costi può essere interpretato come investimento. Questi incontri periodici possono aiutare i team a individuare i KPI basati sul valore in grado di garantire un valore aggiunto all'intera organizzazione.

Risorse

Documenti correlati:

- AWS CCOE Blog (Blog CCOE AWS)
- Creating Cloud Business Office (Creazione di un ufficio aziendale per il cloud)
- CCOE - Cloud Center of Excellence (CCoE - Centro di eccellenza del Cloud)

Video correlati:

- Vanguard CCOE Success Story (Storia di successo CCOE Vanguard)

Esempi correlati:

- Using a Cloud Center of Excellence (CCOE) to Transform the Entire Enterprise (Utilizzo di un Centro di eccellenza del Cloud [CCoE] per trasformare l'intera azienda)
- Building a CCOE to transform the entire enterprise (Creazione di un Centro di eccellenza del Cloud [CCoE] per trasformare l'intera azienda)
- 7 Pitfalls to Avoid When Building CCOE (7 errori da evitare durante la creazione di un Centro di eccellenza del Cloud [CCoE])

COST01-BP02 Definizione di una partnership tra team finanziari e tecnologici

Coinvolgi i team finanziari e tecnologici nelle discussioni su costi e utilizzo in tutte le fasi del tuo approccio al cloud. I team si riuniscono regolarmente e discutono argomenti quali obiettivi e target organizzativi, stato attuale di costi e utilizzo e pratiche finanziarie e contabili.

Livello di rischio associato se questa best practice non fosse adottata: Alto

Guida all'implementazione

i team tecnologici possono innovare più rapidamente nel cloud grazie a cicli di approvazione, approvvigionamento e distribuzione dell'infrastruttura più brevi. Può trattarsi di una novità per le organizzazioni finanziarie che in precedenza erano abituate a eseguire processi dispendiosi, in termini di tempo e di risorse, per acquistare e distribuire capitale in data center e ambienti locali, allocando i costi solo in fase di approvazione del progetto.

Dal punto di vista delle organizzazioni finanziarie e addette all'approvvigionamento, il processo di elaborazione del piano degli investimenti, della richiesta, dell'approvazione e dell'approvvigionamento degli investimenti e dell'installazione dell'infrastruttura fisica è stato interiorizzato e standardizzato da decenni:

- I team di progettazione o IT sono in genere i richiedenti
- I vari team finanziari fungono da approvatori e procuratori
- I team operativi assemblano, implementano e distribuiscono un'infrastruttura pronta all'uso

Con l'adozione del cloud, l'approvvigionamento e il consumo dell'infrastruttura non sono più vincolati da una catena di dipendenze. Nel modello cloud, i team tecnologici e del prodotto non sono più semplici "costruttori", ma anche operatori e proprietari dei loro prodotti, responsabili della maggior parte delle attività storicamente associate ai team finanziari e operativi, compresi l'approvvigionamento e l'implementazione.

Quanto in realtà è necessario per il provisioning delle risorse cloud è un account utente e il set appropriato di autorizzazioni, elementi questi che riducono i rischi IT e finanziari. Ciò significa che ai team basta un numero ridotto di clic o chiamate API per terminare le risorse cloud non necessarie o inattive. Ciò inoltre consente ai team tecnologici di velocizzare l'innovazione, grazie all'agilità e alla capacità di potenziare e quindi ridimensionare i vari progetti sperimentali. Se da un lato la natura variabile del consumo del cloud può influenzare la prevedibilità dal punto di vista del processo di elaborazione del piano degli investimenti e delle previsioni, il cloud fornisce alle organizzazioni la capacità di ridurre il costo del provisioning eccessivo e contemporaneamente il costo delle opportunità associato a un provisioning insufficiente di carattere conservativo.

Stabilisci una collaborazione tra i principali stakeholder finanziari e tecnologici per creare una comprensione condivisa degli obiettivi organizzativi e sviluppare meccanismi che consentano il successo finanziario nel modello di spesa variabile del cloud computing. I team pertinenti all'interno della tua organizzazione devono essere coinvolti nelle discussioni su costi e utilizzo in tutte le fasi del tuo viaggio verso il cloud; tra di essi vi sono:

- Responsabili finanziari: CFO, controllori finanziari, pianificatori finanziari, analisti aziendali, approvvigionamento e selezione delle risorse e contabilità fornitori devono comprendere il modello di consumo del cloud, le opzioni di acquisto e il processo di fatturazione mensile. I team finanziari devono collaborare con i team tecnologici per creare e divulgare a livello aziendale una narrazione del valore IT che aiuti i team aziendali a comprendere lo stretto legame tra spesa in tecnologie e risultati aziendali. In questo modo, la spesa tecnologica viene considerata non tanto come un costo, quanto piuttosto come un vero e proprio investimento. A causa delle differenze fondamentali tra il cloud (ad esempio il tasso di variazione dell'utilizzo, i prezzi a consumo o a scaglioni, i modelli di prezzo e le informazioni dettagliate su fatturazione e utilizzo) e le operazioni in locale, è essenziale che l'organizzazione finanziaria capisca in che modo l'utilizzo del cloud può influire sugli aspetti aziendali, tra cui processi di approvvigionamento, monitoraggio degli incentivi, allocazione dei costi e bilanci.
- Responsabili tecnologici: i responsabili tecnologici (inclusi i proprietari di prodotti e applicazioni) devono essere a conoscenza dei requisiti finanziari (ad esempio i vincoli di budget) e dei requisiti aziendali (ad esempio i contratti sul livello di servizio). In questo modo, il carico di lavoro può essere implementato in modo opportuno per raggiungere gli obiettivi desiderati dall'azienda.

La collaborazione tra finanza e tecnologia offre i seguenti vantaggi:

- I team finanziari e tecnologici hanno una visibilità quasi in tempo reale su costi e utilizzo.
- I team finanziari e tecnologici stabiliscono una procedura operativa standard per gestire le variazioni di spesa nel cloud.
- Gli stakeholder finanziari fungono da consulenti strategici per quanto riguarda il modo in cui il capitale viene utilizzato per acquistare sconti a fronte di impegni (ad esempio, istanze riservate o AWS Savings Plans) e il modo in cui il cloud viene utilizzato per far crescere l'organizzazione.
- I processi di approvvigionamento e di contabilità esistenti vengono applicati al cloud.
- I team finanziari e tecnologici collaborano per prevedere costi e utilizzo di AWS futuri, al fine di allineare e sviluppare i budget aziendali.
- La comunicazione all'interno dell'organizzazione migliora attraverso un linguaggio condiviso e una comprensione comune dei concetti finanziari.

Altri stakeholder all'interno della tua organizzazione che devono essere coinvolti nelle discussioni su costi e utilizzo includono:

- Proprietari delle unità aziendali: i proprietari delle unità aziendali devono comprendere il modello aziendale del cloud in modo da indirizzare l'operato delle unità aziendali e di tutta l'azienda. Questa conoscenza del cloud è fondamentale quando è necessario prevedere la crescita e l'utilizzo del carico di lavoro, ma anche quando si valutano le diverse opzioni di acquisto, come le istanze riservate o i Savings Plans.
- Team di progettazione: lo sviluppo di una partnership tra team finanziari e tecnologici è essenziale per la creazione di una cultura consapevole dei costi che incoraggi il coinvolgimento degli ingegneri nella gestione finanziaria del cloud. Uno dei problemi comuni dei professionisti della gestione finanziaria del cloud o delle operazioni e dei team finanziari è far capire agli ingegneri l'attività nel cloud nel suo complesso e implementare le azioni consigliate.
- Terze parti: se la tua organizzazione si avvale di terze parti (ad esempio, consulenti o strumenti), assicurati che esse siano allineate ai tuoi obiettivi finanziari e possano dimostrare sia l'allineamento, tramite i loro modelli di coinvolgimento, sia il ritorno sull'investimento (ROI). In genere, le terze parti contribuiscono alla creazione di report e all'analisi di eventuali carichi di lavoro da esse gestiti, e forniscono anche l'analisi dei costi relativi ai carichi di lavoro da esse progettati.

L'implementazione della gestione finanziaria del cloud e il conseguimento dei risultati richiedono la stretta collaborazione tra team finanziari, tecnologici e aziendali, nonché un cambiamento nel modo in cui la spesa cloud viene comunicata e valutata all'interno dell'organizzazione. Includi i team di progettazione in modo da renderli partecipi delle discussioni su costi e utilizzi in tutte le fasi e incoraggiali ad attenersi alle best practice e ad adottare le azioni concordate.

Passaggi dell'implementazione

- Definizione dei membri chiave: Verifica che tutti i membri rilevanti dei team finanziari e tecnologici partecipino alla partnership. I membri del team finanziario interessati saranno quelli che hanno a che fare con la fatturazione dei servizi cloud. In genere si tratta di CFO, controllori finanziari, pianificatori finanziari, analisti aziendali, addetti agli acquisti e al sourcing. I membri tecnologici sono in genere i proprietari di prodotti e applicazioni, manager tecnici e rappresentanti di tutti i team che si basano sul cloud. Altri membri possono includere i responsabili di unità aziendali, ad esempio il marketing che influenzerà l'utilizzo dei prodotti, e terze parti, come i consulenti, necessari per garantire l'allineamento agli obiettivi e meccanismi e per fornire assistenza nell'elaborazione dei report.
- Definizione degli argomenti oggetto della discussione: Definisci gli argomenti comuni tra i team o che necessitano di una comprensione condivisa. Segui il costo dal momento in cui viene creato, fino al pagamento della fattura. Prendi nota di tutti i membri coinvolti e dei processi organizzativi

che devono essere applicati. Comprendi ogni fase o processo e le informazioni associate, come i modelli di prezzo disponibili, i prezzi a scaglioni, i modelli di sconto, il budget e i requisiti finanziari.

- Definizione di una regolare cadenza: per creare una partnership tra team finanziari e tecnologici, definisci la periodicità delle comunicazioni per creare e gestire l'allineamento. Il gruppo deve riunirsi regolarmente in base ai propri obiettivi e parametri. Una cadenza tipica implica la revisione dello stato dell'organizzazione, la revisione dei programmi attualmente in esecuzione e la revisione dei parametri finanziari e di ottimizzazione generali. Quindi, per i carichi di lavoro chiave, è opportuno elaborare report più dettagliati.

Risorse

Documenti correlati:

- Blog delle novità di AWS

COST01-BP03 Definizione di budget e previsioni per il cloud

Adatta i processi di previsione e di budgeting organizzativi esistenti in modo che siano compatibili con la natura altamente variabile dei costi e dell'utilizzo del cloud. I processi devono essere dinamici, utilizzando algoritmi basati su tendenze o fattori chiave aziendali o una combinazione di entrambi.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

Nelle tradizionali configurazioni IT on-premises, i clienti spesso devono affrontare la sfida di pianificare i costi fissi che variano solo occasionalmente, di solito con i nuovi acquisti di hardware e servizi IT per soddisfare i picchi di domanda. Cloud AWS adotta invece un approccio diverso, in cui i clienti pagano per le risorse che utilizzano in base alle loro effettive esigenze IT e aziendali. Nell'ambiente cloud, la domanda può variare su base mensile, giornaliera o persino oraria.

Il cloud offre efficienza, velocità e agilità, consolidando un modello di costo e utilizzo altamente variabile. I costi possono diminuire o talvolta aumentare in seguito all'incremento dell'efficienza dei carichi di lavoro o all'implementazione di nuovi carichi di lavoro e funzionalità. Man mano che i carichi di lavoro scalano per servire la clientela in crescita, l'utilizzo e i costi del cloud aumentano di conseguenza a causa del maggiore uso di risorse. Questa flessibilità dei servizi cloud si estende ai costi e alle previsioni, offrendo un certo grado di elasticità.

Per ottenere la pianificazione più accurata possibile, è essenziale allinearsi prontamente a queste mutevoli esigenze aziendali e ai fattori trainanti della domanda. I tradizionali processi di budget dell'organizzazione devono cambiare per far fronte a questa variabilità.

Valuta la modellazione dei costi mentre prevedi la spesa dei nuovi carichi di lavoro. La modellazione dei costi crea una comprensione di base dei costi del cloud previsti che ti consente di calcolare il costo totale di proprietà (TCO), il ritorno sull'investimento (ROI) e altri dati finanziari nonché stabilire obiettivi e aspettative con le parti interessate e identificare le opportunità di ottimizzazione dei costi.

È necessario che l'organizzazione comprenda la definizione dei costi e i raggruppamenti accettati. Il livello di dettaglio usato per le previsioni può variare in base alla struttura dell'organizzazione e ai flussi di lavoro interni. Scegli la granularità adatta ai tuoi requisiti specifici e alla configurazione dell'organizzazione. È importante comprendere a quale livello viene eseguita la previsione:

- Account di gestione o AWS Organizations: l'account di gestione è quello che usi per creare AWS Organizations. Per impostazione predefinita Organizations dispone di un account di gestione.
- Account collegato o membro: un account in Organizations è un Account AWS standard che include le risorse AWS e le identità che possono accedervi.
- Ambiente: un ambiente è una raccolta di risorse AWS che esegue una versione dell'applicazione. È possibile creare un ambiente con più account collegati o membri.
- Progetto: un progetto è una combinazione di attività oppure obiettivi prestabiliti da realizzare entro un determinato periodo di tempo. È importante considerare il ciclo di vita del progetto durante la previsione.
- Servizi AWS: gruppi o categorie, come i servizi di calcolo o archiviazione, in cui puoi raggruppare i servizi AWS per le previsioni.
- Raggruppamento personalizzato: puoi creare gruppi personalizzati in base alle esigenze della tua organizzazione, ad esempio business unit, centri di costo, team, tag di allocazione dei costi, categorie di costo, account collegati oppure una combinazione di questi.

Individua i fattori aziendali che possono influire sui costi di utilizzo e fai le previsioni per ciascuno di essi separatamente per calcolare in anticipo l'utilizzo previsto. Alcuni fattori possono essere collegati ai team IT e di prodotto dell'organizzazione. Altri fattori aziendali, come eventi di marketing, promozioni, espansioni geografiche, fusioni e acquisizioni, sono noti ai responsabili dell’area vendite, marketing e commerciale, quindi è importante collaborare e tenere conto anche di tutti questi fattori trainanti della domanda.

Per ottenere le previsioni relative ai costi per un intervallo di tempo futuro definito in base alle spese pregresse è possibile utilizzare AWS Cost Explorer. Il motore di previsione di AWS Cost Explorer segmenta i dati storici in base ai tipi di addebito, ad esempio le istanze riservate, e utilizza una combinazione di machine learning e modelli basati su regole per elaborare le previsioni di spesa per tutti i singoli tipi di addebito.

Una volta stabilito il processo di previsione e creati i modelli, puoi utilizzare Budget AWS per definire budget personalizzati a livello granulare, specificando il periodo di tempo, la ricorrenza o l'importo (fisso o variabile) e aggiungendo i filtri come servizi, Regione AWS e tag. Il budget è generalmente definito per un solo anno e rimane fisso, richiedendo il rispetto rigoroso di tutte le parti coinvolte. Al contrario, le previsioni sono più flessibili, consentono adattamenti nel corso dell'anno e forniscono proiezioni dinamiche su un periodo di uno, due o tre anni. I budget e le previsioni svolgono un ruolo determinante nella definizione delle aspettative finanziarie tra le varie parti interessate tecnologiche e aziendali. Una previsione e un'implementazione accurate rendono responsabili anche le parti interessate che sono direttamente coinvolte nella gestione dei costi di provisioning e possono aumentare la loro consapevolezza generale dei costi.

Per essere informati sulle prestazioni dei budget esistenti, puoi creare e pianificare report Budget AWS da inviare tramite e-mail alle parti interessate con cadenza regolare. Puoi anche creare avvisi di Budget AWS basati sui costi effettivi, ovvero avvisi intrinsecamente reattivi, oppure sui costi previsti, ovvero avvisi che consentono di implementare tempestivamente azioni correttive a fronte di potenziali eventi di superamento dei costi. Puoi ricevere un avviso quando il costo o l'utilizzo supera un determinato livello oppure si prevede che superi l'importo definito nel budget.

Modifica i processi di budget e previsione esistenti per renderli più dinamici utilizzando gli algoritmi basati sulle tendenze con i costi storici come input e gli algoritmi basati sui fattori aziendali, ad esempio il lancio di nuovi prodotti, l'espansione regionale o i nuovi ambienti per i carichi di lavoro, ideali per un ambiente di spesa dinamico e variabile. Dopo aver determinato la previsione basata sulle tendenze utilizzando Cost Explorer o qualsiasi altro strumento, usa il AWS Pricing Calculator per stimare il caso d'uso AWS e i costi futuri in base all'utilizzo previsto (traffico, richieste al secondo o istanze Amazon EC2 richieste).

Controlla l'accuratezza di questa previsione perché i budget devono essere impostati sulla base di questi calcoli e queste stime. Monitora la precisione e l'efficacia delle previsioni dei costi del cloud integrate. Esamina regolarmente la spesa effettiva rispetto alla tua previsione e apporta le modifiche necessarie per ottenere una maggiore accuratezza. Controlla la varianza prevista ed esegui l'analisi della causa principale della varianza indicata per intervenire e modificare le previsioni.

Come indicato in COST01-BP02 Definizione di una partnership tra team finanziari e tecnologici, è importante favorire la collaborazione e le opportunità di contatto tra IT, finanza e le altre parti interessate per verificare che tutti stiano utilizzando gli stessi strumenti e processi a garanzia del modello di consistenza. Nei casi in cui si rendano necessarie modifiche del budget, l'incremento della frequenza delle occasioni di contatto permette di intervenire e reagire più tempestivamente.

Passaggi dell'implementazione

- Definisci il linguaggio dei costi nell'organizzazione: crea un comune linguaggio dei costi AWS all'interno dell'organizzazione con più dimensioni e raggruppamenti. Assicurati che le parti interessate comprendano la granularità delle previsioni, i modelli di prezzo e il livello delle previsioni dei costi.
- Analizza le previsioni basate sulle tendenze: utilizza gli strumenti per le previsioni basate sulle tendenze come AWS Cost Explorer e Amazon Forecast. Analizza i costi di utilizzo rispetto a più dimensioni, come servizi, account, tag e categorie di costi. Se sono necessarie previsioni avanzate, importa i dati di costi e utilizzo AWS (CUR) in Amazon Forecast, che applica la regressione lineare alla previsione come forma di machine learning.
- Analizza le previsioni basate sui fattori aziendali: determina l'impatto dei fattori aziendali sull'utilizzo del cloud e fai previsioni per ciascuno di essi separatamente per calcolare in anticipo il costo di utilizzo previsto. Collabora a stretto contatto con i responsabili delle business unit e le parti interessate per comprendere l'impatto dei nuovi fattori aziendali e calcolare le variazioni dei costi previste per definire budget accurati.
- Aggiorna i processi di previsione e budget esistenti: a seconda dei metodi di previsione adottati, ad esempio basati sulle tendenze, sui fattori aziendali o su una combinazione di entrambi i metodi, definisci i tuoi processi di previsione e budget. I budget devono essere calcolati, realistici e basati sulle previsioni.
- Configura avvisi e notifiche: utilizza gli avvisi e il rilevamento delle anomalie dei costi di Budget AWS per ricevere avvisi e notifiche.
- Esegui revisioni periodiche con le principali parti interessate: ad esempio, allinea i cambiamenti a livello di direzione e utilizzo aziendale con le parti interessate dei team IT, finanziario e della piattaforma nonché di altre aree aziendali.

Risorse

Documenti correlati:

- AWS Cost Explorer

- AWS Cost and Usage Report
- Forecasting with Cost Explorer
- Amazon QuickSight Forecasting
- Amazon Forecast
- Budget AWS

Video correlati:

- How can I use Budget AWS to track my spending and usage
- AWSCost Optimization Series: Budget AWS

Esempi correlati:

- Understand and build driver-based forecasting
- How to establish and drive a forecasting culture
- How to improve your cloud cost forecasting
- Using the right tools for your cloud cost forecasting

COST01-BP04 Implementazione della consapevolezza dei costi nei processi dell'organizzazione

Implementa la consapevolezza dei costi e crea trasparenza e funzionalità di controllo in processi nuovi o esistenti che influiscono sull'utilizzo e sfrutta i processi esistenti per favorire la consapevolezza dei costi. Implementa la consapevolezza dei costi nella formazione dei dipendenti.

Livello di rischio associato se questa best practice non fosse adottata: Alto

Guida all'implementazione

La consapevolezza dei costi deve essere implementata nei processi organizzativi nuovi ed esistenti. Si tratta di un prerequisito fondamentale per altre best practice. È consigliabile riutilizzare e modificare i processi esistenti, laddove possibile, riducendo al minimo l'impatto sull'agilità e sulla velocità. Comunica i costi del cloud ai team tecnologici e ai responsabili dei processi decisionali nei team aziendali e finanziari per accrescere la consapevolezza dei costi e definisci indicatori chiave delle prestazioni (KPI) per l'efficienza da segnalare alle parti coinvolte nelle varie aree finanziarie e aziendali. Le seguenti raccomandazioni aiuteranno a implementare la consapevolezza dei costi nel carico di lavoro:

- Verifica che la gestione delle modifiche includa una misurazione dei costi per quantificare l'impatto finanziario delle modifiche. Questo aiuta a risolvere in modo proattivo le problematiche relative ai costi nonché a evidenziare i risparmi ottenuti.
- Verifica che l'ottimizzazione dei costi sia un componente fondamentale delle tue capacità operative. Ad esempio, puoi sfruttare gli attuali processi di gestione degli incidenti per analizzare e identificare la causa principale di anomalie di costi e utilizzo o delle eccedenze di costo.
- Accelera la riduzione dei costi e la realizzazione del valore aggiunto attraverso l'automazione o l'utilizzo di strumenti. Quando valuti i costi dell'implementazione, includi nella valutazione un componente ROI per giustificare l'investimento di tempo o denaro.
- Assegna i costi del cloud mediante l'implementazione delle policy di showback/chargeback per la spesa cloud, compresa la spesa per opzioni di acquisto basate su impegno, servizi condivisi e acquisti su marketplace, a supporto di un consumo del cloud maggiormente consapevole dei costi.
- Estendi i programmi di formazione e sviluppo esistenti per includere la formazione sulla consapevolezza dei costi in tutta l'organizzazione, comprese attività di formazione continua e certificazione. In questo modo, creerai un'organizzazione in grado di gestire in modo autonomo i costi e l'utilizzo.
- Sfrutta i vantaggi degli strumenti nativi AWS gratuiti, come AWS Cost Anomaly Detection, Budget AWSe Report Budget AWS.

Quando le organizzazioni adottano in modo sistematico le best practice relative alla gestione finanziaria del cloud, questi comportamenti vengono inglobati nelle procedure di lavoro e nei processi decisionali. Ne risulterà una cultura basata su una maggiore consapevolezza dei costi, condivisa dagli sviluppatori che creano nuove applicazioni per il cloud e dai responsabili dell'area finanziaria che analizzano il ROI per questi nuovi investimenti a livello di cloud.

Passaggi dell'implementazione

- Identificazione dei processi organizzativi pertinenti: Ogni unità organizzativa esamina i propri processi e identifica i processi che influiscono su costi e utilizzo. Tutti i processi che determinano la creazione o la cessazione di una risorsa devono essere inclusi nella revisione. Inoltre, individua i processi che possono supportare la consapevolezza dei costi nella tua azienda, ad esempio la gestione degli incidenti e la formazione.
- Definizione di una cultura consapevole dei costi autosufficiente: assicurati che tutte le parti coinvolte rilevanti siano concordi sulla causa della modifica e sull'impatto come costo in modo che abbiano la piena consapevolezza del costo del cloud. Ciò consentirà all'organizzazione di definire una cultura consapevole dei costi autosufficiente finalizzata all'innovazione.

- Aggiornamento dei processi con la consapevolezza dei costi: Ogni processo viene modificato in modo che ci sia una consapevolezza dei costi. Il processo potrebbe richiedere ulteriori controlli preliminari, ad esempio la valutazione dell'impatto dei costi, oppure controlli successivi che attestino il verificarsi dei cambiamenti previsti in termini di costi e utilizzo. I processi di supporto come la formazione e la gestione degli incidenti possono essere estesi per includere elementi relativi a costi e utilizzo.

Per ottenere assistenza, contatta gli esperti di gestione finanziaria del cloud mediante il team del tuo account oppure esplora le risorse e i documenti correlati elencati di seguito.

Risorse

Documenti correlati:

- Gestione finanziaria del cloud con AWS

Esempi correlati:

- Strategy for Efficient Cloud Cost Management (Strategia per un'efficiente gestione dei costi del cloud)
- Cost Control Blog Series #3: How to Handle Cost Shock (Blog relativo al controllo dei costi - Serie 3: Come gestire l'impatto dei costi)
- A Beginner’s Guide to AWS Cost Management (Guida per principianti alla AWS Cost Management)

COST01-BP05 Invio di report e notifiche sull'ottimizzazione dei costi

Imposta i budget per il cloud e configura i meccanismi per rilevare anomalie nell'utilizzo. Configura gli strumenti correlati per ricevere avvisi su costi e utilizzo rispetto a obiettivi predefiniti e ricevi notifiche quando l'utilizzo supera tali obiettivi. Organizza riunioni regolari per analizzare l'economicità dei tuoi carichi di lavoro e promuovere la consapevolezza dei costi.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

È necessario rendicontare regolarmente l'ottimizzazione dei costi e dell'utilizzo all'interno dell'organizzazione. Puoi implementare sessioni dedicate per discutere le prestazioni in termini di costi o includere l'ottimizzazione dei costi nei regolari cicli di reporting operativi per i tuoi carichi di

lavoro. Utilizza servizi e strumenti per monitorare regolarmente le prestazioni in termini di costi e implementare opportunità di risparmio sui costi.

Visualizza i costi e l'utilizzo con più filtri e granularità utilizzando AWS Cost Explorer, che fornisce dashboard e report come i costi per servizio o per account, i costi giornalieri o i costi del marketplace. Monitora l'avanzamento di costi e utilizzo rispetto ai budget configurati attraverso i report Budget AWS.

Utilizza Budget AWS per configurare budget personalizzati al fine di tenere traccia dei costi e dell'utilizzo e reagire con tempestività agli avvisi ricevuti via e-mail o alle notifiche Amazon Simple Notification Service (Amazon SNS) in caso di superamento della soglia definita. Imposta il periodo di budget preferito su giornaliero, mensile, trimestrale o annuale e crea limiti di budget specifici per essere costantemente informato sui valori di utilizzo e sui costi effettivi o previsti rispetto alla soglia definita per il budget. Puoi anche configurare avvisi e operazioni da eseguire automaticamente o in base a un processo di approvazione a fronte di tali avvisi quando viene superato l'obiettivo del budget.

Implementa notifiche su costi e utilizzo in modo che si possa intervenire rapidamente in caso di variazioni impreviste di costi e utilizzo. AWS Cost Anomaly Detection consente di ridurre gli inconvenienti a livello di costi e migliorare il controllo senza rallentare il processo di innovazione. AWS Cost Anomaly Detection individua le spese anomale e le cause principali a favore della riduzione del rischio di imprevisti a livello di fatturazione. Grazie a tre semplici passaggi, puoi creare una funzione di controllo contestualizzato personalizzato e ricevere avvisi quando viene rilevata una spesa anomala.

Puoi anche utilizzare Amazon QuickSight con dati AWS Cost and Usage Report (CUR) per fornire funzionalità di reporting personalizzate con dati più granulari. Amazon QuickSight consente di programmare i report e ricevere via e-mail report periodici sui costi relativi all'utilizzo e sui costi storici o sulle opportunità di riduzione dei costi. Controlla il nostro pannello di controllo Intelligence costi Soluzione (CID) creata su Amazon QuickSight, che offre una visibilità avanzata.

utilizza AWS Trusted Advisor, che mette a disposizione linee guida per verificare se le risorse allocate sono conformi alle best practice AWS in relazione all'ottimizzazione dei costi.

Controlla le tue raccomandazioni Savings Plans tramite grafici visivi confrontandoli con i costi e l'utilizzo granulari. I grafici orari mostrano la spesa on demand insieme all'impegno verso i Savings Plans consigliati, fornendo informazioni sui risparmi stimati, sulla copertura dei Savings Plans e sull'utilizzo dei Savings Plans. Questo aiuta le organizzazioni a capire in che modo i loro Savings

Plans si applicano a ogni ora di spesa senza dover investire tempo e risorse nella creazione di modelli per analizzare la spesa stessa.

Crea periodicamente report contenenti informazioni di primo piano relative a Savings Plans, istanze riservate e suggerimenti per il corretto dimensionamento di Amazon EC2 forniti da AWS Cost Explorer per favorire la riduzione dei costi associati a carichi di lavoro con stato stazionario e a risorse inattive e sottoutilizzate. Individua e ammortizza la spesa associata all'utilizzo non ottimale del cloud relativamente alle risorse implementate. Con utilizzo non ottimale del cloud si intende la creazione di risorse dimensioni errate oppure la presenza di modelli di utilizzo del cloud diversi da quanto previsto. Segui le migliori pratiche di AWS per ridurre gli sprechi o chiedi al tuo account, al team e al partner di aiutarti ottimizzare e ridurre i costi del cloud.

Genera regolarmente report per migliorare le opzioni di acquisto delle risorse al fine di ridurre il costo unitario dei carichi di lavoro. Le opzioni di acquisto quali, ad esempio, Savings Plans, istanze riservate o istanze spot Amazon EC2, offrono il massimo risparmio sui costi per carichi di lavoro con tolleranza ai guasti, consentendo alle parti interessate (responsabili di attività aziendali, team finanziari e tecnologici) di essere coinvolte nelle discussioni di merito.

Condividi i report contenenti opportunità o annunci di nuovi rilasci a supporto della riduzione del costo totale di proprietà (TCO) del cloud. Adotta nuovi servizi, regioni, funzionalità, soluzioni o nuovi modi per migliorare ulteriormente la riduzione dei costi.

Passaggi dell'implementazione

- Configura Budget AWS: Configura Budget AWS su tutti gli account per il tuo carico di lavoro. Imposta un budget per la spesa complessiva dell'account e un budget per il carico di lavoro utilizzando i tag.
  - Well-Architected Labs: utilizzo di costi e governance
- Report sull'ottimizzazione dei costi: Configura un ciclo regolare per discutere e analizzare l'efficienza del carico di lavoro. Utilizzando i parametri stabiliti, segnala i parametri raggiunti e il costo sostenuto per ottenerli. Identifica e correggi eventuali tendenze negative e individua tendenze positive che puoi favorire in tutta l'organizzazione. La rendicontazione dovrebbe coinvolgere i rappresentanti dei team e dei responsabili delle applicazioni, dei responsabili finanziari e dei principali responsabili delle decisioni in merito alla spesa per il cloud.

Risorse

Documenti correlati:

- AWS Cost Explorer
- AWS Trusted Advisor
- Budget AWS
- AWS Cost and Usage Report
- Budget AWS Best Practices (Best practice per Budget AWS)
- Amazon S3 Analytics (Analisi Amazon S3)

Esempi correlati:

- Well-Architected Labs: utilizzo di costi e governance
- Key ways to start optimizing your AWS cloud costs (Principali soluzioni per iniziare a ottimizzare i costi del cloud AWS)

COST01-BP06 Monitoraggio proattivo dei costi

Implementa strumenti e pannelli di controllo per monitorare i costi in modo proattivo per il carico di lavoro. Rivedi regolarmente i costi utilizzando strumenti configurati o pronti all'uso e non limitarti a guardare solo i costi e le categorie quando ricevi le notifiche. Il monitoraggio e l'analisi proattivi dei costi aiutano a individuare i trend positivi e ti consente di promuoverli all'interno dell'organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

si consiglia di monitorare i costi e l'utilizzo all'interno dell'organizzazione in modo proattivo, e non solo in caso di eccezioni o anomalie. I pannelli di controllo con un'elevata visibilità in tutto l'ufficio o l'ambiente di lavoro garantiscono che le persone chiave abbiano accesso alle informazioni di cui hanno bisogno e dimostrano l'attenzione che l'organizzazione presta all'ottimizzazione dei costi. I pannelli di controllo visibili consentono di promuovere attivamente i risultati positivi e di implementarli in tutta l'organizzazione.

Crea procedure giornaliere o frequenti che utilizzino AWS Cost Explorer o qualsiasi altro pannello di controllo, come Amazon QuickSight , per verificare i costi e analizzarli in modo proattivo. Analizza l'utilizzo e i costi dei servizi AWS a livello di account AWS, carico di lavoro o servizio AWS specifico in gruppo o mediante filtri e verifica che siano in linea con quanto previsto. Utilizza tag e granularità a livello orario o di risorsa per filtrare e individuare i costi ricorrenti relativi alle risorse di maggiore utilizzo. Puoi anche creare report personalizzati con il pannello di controllo Intelligence costi, una

soluzione Amazon QuickSight sviluppata dagli AWS Solutions Architect, e confrontare i budget con i costi e l'utilizzo effettivi.

Passaggi dell'implementazione

- Report sull'ottimizzazione dei costi: Configura un ciclo regolare per discutere e analizzare l'efficienza del carico di lavoro. Utilizzando i parametri stabiliti, segnala i parametri ottenuti e il costo sostenuto per ottenerli. Identifica e correggi eventuali tendenze negative e identifica le tendenze positive che puoi favorire in tutta l'organizzazione. L'elaborazione dei report deve coinvolgere i rappresentanti dei team applicativi e dei proprietari, dei team finanziari e di gestione.
- Creazione e abilitazione di Budget AWS con granularità giornaliera relativi a costi e utilizzo per adottare misure tempestive volte a impedire potenziali superamenti dei costi: Budget AWS consente di configurare notifiche di avviso per essere sempre informati se qualsiasi tipo di budget non è conforme alle soglie preconfigurate. Il modo migliore per utilizzare Budget AWS è configurare i costi e l'utilizzo previsti come limite in modo tale che qualsiasi superamento del budget possa essere considerato un superamento del limite di spesa.
- Creazione del AWS Cost Anomaly Detection per il monitoraggio dei costi: AWS Cost Anomaly Detection utilizza la tecnologia avanzata di machine learning per individuare le spese anomale e le cause principali in modo da garantire un intervento tempestivo. Ti consente di configurare funzionalità di monitoraggio dei costi che definiscono i segmenti di spesa da valutare, ad esempio singoli servizi AWS, account membro, tag di allocazione dei costi e categorie di costo, nonché di impostare quando, dove e come riceverai le notifiche di avviso. Per ciascuna funzionalità di monitoraggio, puoi associare più sottoscrizioni agli avvisi per proprietari di azienda e team tecnologici, inclusi un nome, una soglia relativa all'impatto dei costi e la frequenza di avviso (avvisi singoli, riepilogo giornaliero, riepilogo settimanale) per ciascuna sottoscrizione.
- Utilizzo di AWS Cost Explorer o integrazione dei dati AWS Cost and Usage Report (CUR) con i pannelli di controllo Amazon QuickSight per la visualizzazione dei costi dell'organizzazione: La funzionalità AWS Cost Explorer è caratterizzata da un'interfaccia di semplice utilizzo che consente di visualizzare, analizzare e gestire l'utilizzo e i costi AWS nel tempo. Il pannello di controllo Intelligence costi è personalizzabile e accessibile e consente di creare le basi di uno strumento di gestione e ottimizzazione dei costi personalizzato.

Risorse

Documenti correlati:

- Budget AWS

- AWS Cost Explorer
- Daily Cost and Usage Budgets (Budget per costi e utilizzo giornalieri)
- AWS Cost Anomaly Detection

Esempi correlati:

- AWS Well-Architected Labs: visualizzazione
- AWS Well-Architected Labs: visualizzazione avanzata
- Well-Architected Labs: Cloud Intelligence Dashboards (Pannelli di controllo Intelligence cloud)
- Well-Architected Labs: Cost Visualization (Visualizzazione dei costi)
- AWS Cost Anomaly Detection Alert with Slack (Avvisi AWS Cost Anomaly Detection con Slack)

COST01-BP07 Mantenimento dell'aggiornamento sulle nuove versioni dei servizi

Consultati regolarmente con gli esperti o con i partner AWS per valutare quali servizi e caratteristiche offrono un costo inferiore. Consulta i blog AWS e altre fonti di informazione.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

AWS continua ad aggiungere nuove caratteristiche in modo da consentirti di utilizzare le tecnologie più aggiornate a supporto di un più rapido processo di sperimentazione e innovazione. Potresti essere in grado di implementare nuovi servizi e funzionalità AWS per aumentare l'efficienza in termini di costi del carico di lavoro. Consulta regolarmente la pagina Gestione dei costi AWS, il Blog delle novità di AWS, il Blog sulla gestione dei costi AWSe Novità di AWS per informazioni su nuovi servizi e versioni di funzionalità. I post nella sezione Novità forniscono una breve panoramica di tutti gli annunci relativi a servizi AWS, funzionalità ed espansione delle regioni al momento del loro rilascio.

Passaggi dell'implementazione

- Iscriviti ai blog: Vai alle pagine dei blog AWS e iscriviti al Blog delle novità e ad altri blog di interesse. Puoi effettuare la registrazione nella pagina delle preferenze di comunicazione utilizzando il tuo indirizzo e-mail.
- Iscriviti alle novità di AWS: consulta regolarmente il Blog delle novità di AWS e Novità di AWS per informazioni su nuovi servizi e versioni di funzionalità. Iscriviti ai feed RSS oppure utilizza il tuo indirizzo e-mail per essere sempre aggiornato su annunci e nuovi rilasci.

- Segui le informazioni riportate nella sezione relativa alle riduzioni di prezzo AWS: con regolari riduzioni di prezzo su tutti i nostri servizi, AWS ha regolarmente offerto una maggiore efficienza economica ai nostri clienti acquisiti. Ad aprile 2024, AWS ha ridotto i prezzi 115 volte dal suo lancio nel 2006. Se hai ancora qualche dubbio in merito a decisioni commerciali da prendere a causa di questioni relative ai prezzi, puoi fare riferimento ai nuovi tariffari, che includono riduzioni dei prezzi e nuove integrazioni dei servizi. Puoi avere ulteriori informazioni sulle precedenti riduzioni dei prezzi, comprese quelle relative alle istanze Amazon Elastic Compute Cloud (Amazon EC2), nella categoria relativa alla riduzione dei prezzi del Blog delle novità di AWS.
- Eventi e incontri AWS: Partecipa al summit AWS locale e a qualsiasi incontro locale con altre organizzazioni della tua area. Se non riesci a partecipare dal vivo, prova ad accedere agli eventi virtuali per poter ascoltare gli esperti AWS e rimanere informato sui casi aziendali di altri clienti.
- Organizza riunioni con il team del tuo account: Pianifica una cadenza regolare di incontri con il team del tuo account, organizza riunioni con il team e discuti delle tendenze del settore e dei servizi AWS. Parla con gli account manager, i solutions architect e i team di supporto a te assegnati.

Risorse

Documenti correlati:

- Gestione dei costi AWS
- Novità di AWS
- Blog delle novità di AWS

Esempi correlati:

- Amazon EC2 – 15 Years of Optimizing and Saving Your IT Costs (15 anni di ottimizzazione e risparmio dei costi IT)
- AWS News Blog - Price Reduction (Blog delle novità di AWS - Riduzione dei prezzi)

COST01-BP08 Creazione di una cultura consapevole dei costi

Implementa modifiche o programmi all'interno dell'organizzazione per creare una cultura consapevole dei costi. Si consiglia di iniziare in piccolo, per poi implementare programmi di grandi dimensioni e di vasta portata all'aumentare delle capacità e dell'utilizzo del cloud da parte dell'organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: Basso

Guida all'implementazione

Una cultura consapevole dei costi consente di ricalibrare l'ottimizzazione e la gestione finanziaria del cloud (team operativi e finanziari, Centro di eccellenza del Cloud, operazioni nel cloud e così via) attraverso best practice eseguite in modo organico e decentralizzato all'interno di tutta l'organizzazione. La consapevolezza dei costi crea livelli elevati di capacità all'interno dell'organizzazione con uno sforzo minimo, qualcosa di analogo a un approccio centralizzato e dall'alto verso il basso.

La creazione della consapevolezza dei costi nel cloud computing, soprattutto per quanto riguarda i principali driver dei costi, consente ai team di avere la piena consapevolezza dei risultati previsti associati a qualsiasi variazione a livello di costi. I team con accesso agli ambienti cloud devono conoscere i modelli dei prezzi e la differenza tra i tradizionali data center on-premise e il cloud computing.

Il principale vantaggio di una cultura consapevole dei costi è che i team tecnologici ottimizzano i costi in modo proattivo e continuativo (ad esempio, i costi vengono considerati un requisito non funzionale durante la definizione dell'architettura dei nuovi carichi di lavoro oppure quando vengono apportate modifiche ai carichi di lavoro esistenti) anziché eseguire ottimizzazioni reattive dei costi, in caso di necessità.

Piccoli cambiamenti nella cultura possono avere un grande impatto sull'efficienza dei carichi di lavoro attuali e futuri. Esempi di questo tipo includono:

- Avere visibilità e consapevolezza consente ai team tecnici di progettazione di controllare il loro operato e di capire il tipo di impatto che la loro attività ha in termini di costi.
- Gamificare costi e utilizzo in tutta l'organizzazione. Questa operazione può essere eseguita tramite un pannello di controllo visibile pubblicamente o un report che confronta i costi e l'utilizzo normalizzati tra i team (ad esempio, i costi per carico di lavoro e i costi per transazione).
- Premiare l'efficienza dei costi. Ricompensa pubblicamente o privatamente i risultati di ottimizzazione dei costi volontari o non sollecitati e impara dagli errori per evitare di ripeterli in futuro.
- Crea requisiti organizzativi dall'alto verso il basso affinché i carichi di lavoro siano eseguiti nel rispetto dei budget predefiniti.
- Esegui una verifica continua dei requisiti aziendali relativi alle modifiche e dell'impatto dei costi delle modifiche richieste sull'infrastruttura dell'architettura o sulla configurazione del carico di lavoro per essere sicuro di pagare solo quanto è necessario.

- Verifica che il responsabile delle modifiche sia consapevole delle modifiche previste con un impatto sui costi, che a loro volta devono essere confermate dalle parti coinvolte al fine di ottenere risultati aziendali in modo economicamente conveniente.

Passaggi dell'implementazione

- Comunica i costi del cloud ai team tecnologici: per favorire la consapevolezza dei costi e definire indicatori KPI relativi all'efficienza per le parti coinvolte nelle aree finanziarie e aziendali.
- Comunica le modifiche pianificate alle parti coinvolte o ai membri dei team: crea una voce nel programma per discutere le modifiche pianificate e l'impatto costi/benefici a livello di carico di lavoro durante le riunioni settimanali.
- Organizza riunioni con il team del tuo account: definisci una cadenza regolare per le riunioni con il team del tuo account e discuti delle tendenze del settore e dei servizi AWS. Parla con account manager, architect e team di supporto a te assegnati.
- Condividi le storie di successo: condividi le storie di successo relative alla riduzione dei costi per qualsiasi carico di lavoro, Account AWS o organizzazione per creare un atteggiamento favorevole e incoraggiare la consapevolezza a questo proposito.
- Formazione: assicurati che i team tecnici o i membri dei vari team abbiano ricevuto una formazione adeguata in merito alla consapevolezza dei costi delle risorse nel Cloud AWS.
- Eventi e incontri AWS: partecipa al summit AWS locale e a qualsiasi incontro locale con altre organizzazioni della tua area.
- Iscriviti ai blog: Vai alle pagine dei blog AWS e iscriviti al Blog delle novità e altri blog rilevanti per essere sempre aggiornato sulle nuove versioni, implementazioni, esempi e modifiche condivise da AWS.

Risorse

Documenti correlati:

- Blog AWS
- Gestione dei costi AWS
- Blog delle novità di AWS

Esempi correlati:

- Gestione finanziaria del cloud con AWS
- AWS Well-Architected Labs: Cloud Financial Management (Gestione finanziaria del cloud)

COST01-BP09 Quantificare il valore aggiunto realizzato attraverso l'ottimizzazione dei costi

La quantificazione del valore aggiunto realizzato tramite l'ottimizzazione dei costi consente di comprendere l'intero set di vantaggi per la tua organizzazione. Poiché l'ottimizzazione dei costi è un investimento necessario, la quantificazione del valore aggiunto consente di spiegare il ritorno sull'investimento agli stakeholder. La quantificazione del valore aggiunto può aiutarti a ottenere maggiori consensi dagli stakeholder sugli investimenti futuri in materia di ottimizzazione dei costi, e fornisce un framework per misurare i risultati delle attività di ottimizzazione dei costi della tua organizzazione.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Quantificare il valore aziendale significa misurare i vantaggi che le aziende ottengono dalle azioni e dalle decisioni che prendono. Il valore aziendale può essere tangibile (riduzione delle spese o aumento dei profitti) o intangibile (migliore reputazione del marchio o maggiore soddisfazione del cliente).

Quantificare il valore aziendale derivante dall'ottimizzazione dei costi significa determinare il valore o i vantaggi ottenuti dall'impegno dedicato a rendere più efficiente la spesa. Ad esempio, supponiamo che un'azienda spenda 100.000 dollari per implementare un carico di lavoro su AWS e successivamente lo ottimizzi, portandone il costo a 80.000 dollari senza sacrificare la qualità o la produzione. In questo scenario, il valore aziendale quantificato derivante dall'ottimizzazione dei costi è un risparmio di 20.000 dollari. Ma oltre ai semplici risparmi, l'azienda potrebbe anche quantificare il valore in termini di tempi di consegna più rapidi, maggiore soddisfazione dei clienti o altre metriche derivanti dall'impegno nell'ambito dell'ottimizzazione dei costi. Le parti interessate devono prendere decisioni in merito al potenziale valore dell'ottimizzazione dei costi, al costo dell'ottimizzazione del carico di lavoro e al valore del ritorno sugli investimenti.

oltre a rendicontare i risparmi derivanti dall'ottimizzazione dei costi, è consigliabile quantificare il valore aggiunto fornito. I vantaggi dell'ottimizzazione dei costi sono in genere quantificati in termini di costi inferiori per ottenere un risultato aziendale. Ad esempio, puoi quantificare la riduzione dei costi di Amazon Elastic Compute Cloud (Amazon EC2) quando acquisti Savings Plans, che riduce i costi e mantiene i livelli di output del carico di lavoro. Puoi quantificare la riduzione dei costi di AWS

quando le istanze Amazon EC2 inattive vengono rimosse o quando i volumi Amazon Elastic Block Store (Amazon EBS) scollegati vengono eliminati.

I vantaggi derivanti dall'ottimizzazione dei costi, tuttavia, vanno oltre la riduzione o l'eliminazione dei costi. Prendi in considerazione l'acquisizione di dati aggiuntivi per misurare i miglioramenti dell'efficienza e il valore aggiunto.

Passaggi dell'implementazione

- Valuta i vantaggi aziendali: questo è il processo di analisi e regolazione dei costi del Cloud AWS in modo da massimizzare i vantaggi derivanti da ogni dollaro speso. Invece di concentrarti sulla riduzione dei costi senza considerare il valore aziendale, nell'ambito dell'ottimizzazione dei costi valuta i vantaggi aziendali e il ritorno sugli investimenti, che potrebbero aumentare il valore del denaro speso. Si tratta di spendere con saggezza e di fare investimenti e spese nelle aree che producono i migliori rendimenti.
- Analizza la previsione dei costi AWS: la previsione consente agli stakeholder finanziari di stabilire le aspettative con altri soggetti interni ed esterni dell'organizzazione e aiuta a migliorare la prevedibilità finanziaria dell'organizzazione. AWS Cost Explorer può essere utilizzato per effettuare previsioni sui costi e sull'utilizzo.

Risorse

Documenti correlati:

- Vantaggi economici del Cloud AWS
- Blog AWS
- Gestione dei costi AWS
- AWS News Blog
- whitepaper sul principio dell'affidabilità secondo il Canone di architettura
- Esploratore dei costi AWS

Video correlati:

- Unlock Business Value with Windows on AWS

Esempi correlati:

- Measuring and Maximizing the Business Value of Customer 360
- The Business Value of Adopting Amazon Web Services Managed Databases
- The Business Value of Amazon Web Services for Independent Software Vendors
- Business Value of Cloud Modernization
- The Business Value of Migration to Amazon Web Services

Comprensione delle spese e dell'utilizzo

Domande

- COST 2. In che modo gestisci l'utilizzo?
- COST 3. In che modo monitori i costi e l'utilizzo?
- COST 4. In che modo disattivi le risorse?

COST 2. In che modo gestisci l'utilizzo?

Stabilisci policy e meccanismi per verificare che i costi sostenuti mentre raggiungi gli obiettivi siano adeguati. Utilizzando un approccio di controllo e bilanciamento reciproco, è possibile innovare senza spendere troppo.

Best practice

- COST02-BP01 Sviluppo di policy basate sui requisiti dell'organizzazione
- COST02-BP02 Implementazione di obiettivi e target
- COST02-BP03 Implementazione di una struttura di account
- COST02-BP04 Implementazione di gruppi e ruoli
- COST02-BP05 Implementazione dei controlli di costo
- COST02-BP06 Monitoraggio del ciclo di vita del progetto

COST02-BP01 Sviluppo di policy basate sui requisiti dell'organizzazione

Sviluppa policy che definiscano il modo in cui le risorse vengono gestite dalla tua organizzazione e controllale periodicamente. Le policy devono coprire gli aspetti dei costi relativi alle risorse e ai carichi di lavoro, comprese la creazione, la modifica e la disattivazione nel ciclo di vita delle risorse.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Comprendere i costi e i fattori chiave della tua organizzazione è fondamentale per gestire i costi e l'utilizzo in modo efficiente e per identificare le opportunità di riduzione dei costi. In genere, le organizzazioni gestiscono molteplici carichi di lavoro eseguiti da più team. Questi team possono trovarsi in diverse unità aziendali, ognuna con un proprio flusso di ricavi. La capacità di attribuire i costi delle risorse ai singoli proprietari del carico di lavoro, del prodotto o dell'organizzazione incoraggia un comportamento di utilizzo efficiente e contribuisce a ridurre gli sprechi. Un monitoraggio accurato dei costi e dell'utilizzo consente di comprendere quanto sia ottimizzato un carico di lavoro e quanto siano redditizi i prodotti e le unità organizzative. Questa conoscenza consente di prendere decisioni più informate su dove allocare le risorse all'interno dell'organizzazione. La consapevolezza dell'utilizzo a tutti i livelli dell'organizzazione è fondamentale per promuovere il cambiamento, poiché la modifica dell'utilizzo determina variazioni dei costi. Prova a adottare una strategia versatile per acquisire consapevolezza delle tue spese.

il primo passo per attuare la governance consiste nell'utilizzare i requisiti della tua organizzazione per sviluppare politiche per l'utilizzo del cloud. Queste policy definiscono il modo in cui l'organizzazione utilizza il cloud e il modo in cui le risorse vengono gestite. Le policy devono coprire tutti gli aspetti dei costi relativi alle risorse e ai carichi di lavoro correlati a costi o utilizzo, compresa la creazione, la modifica e la disattivazione durante il ciclo di vita di una risorsa. Verifica che policy e procedure vengano eseguite e implementate per qualsiasi modifica apportata in un ambiente cloud. Durante gli incontri per la gestione delle modifiche IT, poni domande relative all'impatto sui costi delle modifiche pianificate (se implicano un aumento o una riduzione), alla giustificazione aziendale e ai risultati attesi.

Le policy devono essere semplici, in modo che siano facilmente comprensibili e possano essere implementate in modo efficace in tutta l'organizzazione. Le policy devono anche essere facili da seguire e interpretare (in modo da essere utilizzate) e specifiche (senza interpretazioni errate tra i team). Inoltre, devono essere ispezionate periodicamente (come i nostri meccanismi) e aggiornate man mano che le condizioni o le priorità aziendali dei clienti cambiano, il che renderebbe la policy obsoleta.

Inizia con policy ampie e di alto livello, ad esempio in quale regione geografica è consentito l'utilizzo o l'ora del giorno in cui le risorse devono essere in esecuzione. Raffina gradualmente le policy per le varie unità organizzative e i diversi carichi di lavoro. Le policy comuni includono i servizi e le funzionalità che possono essere utilizzati (ad esempio, archiviazione dalle prestazioni inferiori negli ambienti di test e sviluppo), i tipi di risorse che possono essere utilizzati dai diversi gruppi (ad esempio, le dimensioni massime di una risorsa in un account di sviluppo possono essere impostate

su medie) e per quanto tempo queste risorse saranno in uso (temporaneamente, a breve termine o per un periodo di tempo specifico).

Esempio di policy

Di seguito è riportato un esempio di policy che puoi esaminare per creare le tue policy di governance del cloud, basate sull'ottimizzazione dei costi. Assicurati di adattare la policy ai requisiti della tua organizzazione e alle richieste delle parti interessate.

- Nome della policy: definisci un nome chiaro per la policy, ad esempio Ottimizzazione delle risorse e Policy di riduzione dei costi.
- Scopo: spiega perché questa policy dovrebbe essere utilizzata e qual è il risultato previsto. L'obiettivo di questa policy è verificare che sia richiesto un costo minimo per implementare ed eseguire il carico di lavoro desiderato per soddisfare i requisiti aziendali.
- Ambito di applicazione: definisci chiaramente chi deve utilizzare questa policy e quando deve essere utilizzata, ad esempio Team DevOps X per utilizzare questa policy per i clienti nella zona di disponibilità Stati Uniti-Est per l'ambiente X (di produzione o non di produzione).

Dichiarazione delle policy

1. Seleziona us-east-1 o più regioni Stati Uniti-Est in base all'ambiente del carico di lavoro e ai requisiti aziendali (sviluppo, test di accettazione da parte degli utenti, preproduzione o produzione).
2. Pianifica l'esecuzione delle istanze Amazon EC2 e Amazon RDS tra le sei del mattino e le otto di sera (Ora solare orientale [EST]).
3. Arresta tutte le istanze Amazon EC2 inutilizzate dopo otto ore e le istanze Amazon RDS inutilizzate dopo 24 ore di inattività.
4. Interrompi tutte le istanze Amazon EC2 inutilizzate dopo 24 ore di inattività in ambienti non di produzione. Ricorda al proprietario dell'istanza Amazon EC2 (in base ai tag) di esaminare le istanze Amazon EC2 arrestate in produzione e di informarlo che le istanze Amazon EC2 verranno terminate entro 72 ore se non vengono utilizzate.
5. Usa la famiglia e le dimensioni delle istanze generiche come m5.large, quindi ridimensiona l'istanza in base all'utilizzo della CPU e della memoria mediante AWS Compute Optimizer.
6. Assegna la priorità utilizzando il dimensionamento automatico per regolare dinamicamente il numero di istanze in esecuzione in base al traffico.
7. Usa le istanze spot per carichi di lavoro non critici.

8. Esamina i requisiti di capacità per impegnare piani di risparmio o istanze riservate per carichi di lavoro prevedibili e informa il team della gestione finanziaria del cloud.
9. Utilizza le policy Amazon S3 del ciclo di vita per spostare i dati a cui si accede di rado su livelli di archiviazione più economici. Se non è stata definita alcuna policy di conservazione, utilizza Piano intelligente Amazon S3 per spostare automaticamente gli oggetti nel livello archiviato.
10. Monitora l'utilizzo delle risorse e imposta allarmi per attivare eventi di dimensionamento utilizzando Amazon CloudWatch.
11. Per ogni Account AWS, utilizza Budget AWS per impostare i budget di costo e utilizzo per il tuo account in base al centro di costo e alle unità aziendali.
12. L'utilizzo di Budget AWS per impostare i budget di costi e utilizzo del tuo account può aiutarti a tenere sotto controllo le spese ed evitare fatture impreviste, consentendoti di controllare meglio i costi.

Procedura: fornisci procedure dettagliate per l'attuazione di questa policy o fai riferimento ad altri documenti che descrivono come implementare ciascuna dichiarazione della policy. Questa sezione dovrebbe fornire istruzioni dettagliate per l'adempimento dei requisiti della policy.

Per implementare questa policy, puoi utilizzare vari strumenti o regole AWS Config di terze parti per verificare la conformità alla dichiarazione e attivare azioni correttive automatiche utilizzando le funzioni AWS Lambda. Puoi anche usare AWS Organizations per applicare la policy. Inoltre, dovresti controllare regolarmente l'utilizzo delle risorse e modificare la policy, se necessario, per verificare che continui a soddisfare le esigenze aziendali.

Passaggi dell'implementazione

- Incontra le parti interessate: per sviluppare le policy, chiedi alle parti interessate (ufficio aziendale per il cloud, ingegneri o responsabili delle decisioni funzionali per l'applicazione delle policy) all'interno della tua organizzazione di specificare i loro requisiti e documentarli. Segui un approccio iterativo iniziando in modo generale e perfezionando continuamente le unità più piccole in ogni fase. I membri del team includono quelli con interesse diretto nel carico di lavoro, ad esempio unità organizzative o proprietari di applicazioni, nonché gruppi di supporto, come i team di sicurezza e i team finanziari.
- Ottieni conferma: verifica che i team siano d'accordo sulle policy a cui possono accedere e che possono distribuire sul Cloud AWS. Verifica che rispettino le policy della tua organizzazione e conferma che le creazioni di risorse siano in linea con le policy e le procedure concordate.

- Organizza sessioni di formazione per l'onboarding: chiedi ai nuovi membri dell'organizzazione di partecipare a corsi di formazione di onboarding per sviluppare una consapevolezza sui costi e sui requisiti aziendali. Potrebbero adottare policy diverse legate all'esperienza precedente o non rifletterci affatto.
- Definizione delle posizioni per il carico di lavoro: Definisci dove opera il carico di lavoro, incluso il paese e l'area all'interno del paese. Queste informazioni vengono utilizzate per la mappatura su Regioni AWS e sulle zone di disponibilità.
- Definizione e raggruppamento di servizi e risorse: Definisci i servizi richiesti dai carichi di lavoro. Per ogni servizio, specifica i tipi, la dimensione e il numero di risorse richieste. Definisci i gruppi per le risorse in base alla funzione, ad esempio i server di applicazioni o lo storage di database. Le risorse possono appartenere a più gruppi.
- Definizione e raggruppamento degli utenti per funzione: Definisci gli utenti che interagiscono con il carico di lavoro, concentrandoti su ciò che fanno e su come utilizzano il carico di lavoro, non su chi sono o sulla loro posizione nell'organizzazione. Raggruppa utenti o funzioni simili. Puoi utilizzare le policy gestite da AWS come guida di riferimento.
- Definizione delle operazioni: Utilizzando le posizioni, le risorse e gli utenti identificati in precedenza, definisci le azioni richieste da ciascuno di essi per ottenere i risultati del carico di lavoro durante il ciclo di vita (sviluppo, funzionamento e disattivazione). Identifica le operazioni in base ai gruppi, non ai singoli elementi nei gruppi, in ogni posizione. Inizia in generale con lettura o scrittura, quindi perfeziona le azioni specifiche per ciascun servizio.
- Definizione del periodo di revisione: I carichi di lavoro e i requisiti organizzativi possono cambiare nel corso del tempo. Definisci la pianificazione della revisione del carico di lavoro per assicurarti che sia allineata alle priorità organizzative.
- Documentazione delle policy: verifica che le policy definite siano accessibili come richiesto dall'organizzazione. Queste policy vengono utilizzate per implementare, mantenere e controllare l'accesso agli ambienti.

Risorse

Documenti correlati:

- Gestione delle modifiche nel cloud
- Policy gestite da AWS per le funzioni lavorative
- Strategia di fatturazione con account multipli di AWS
- Operazioni, risorse e chiavi di condizione per i servizi AWS

- Gestione e governance AWS
- Controllo dell'accesso a Regioni AWS utilizzando le policy IAM
- Regioni e zone di disponibilità dell'infrastruttura globale

Video correlati:

- AWS Management and Governance at Scale (Gestione e governance AWS su scala)

Esempi correlati:

- VMware - Quali sono le policy cloud?

COST02-BP02 Implementazione di obiettivi e target

Implementa obiettivi e target di costi e utilizzo per il carico di lavoro. Gli obiettivi forniscono indicazioni alla tua organizzazione sui risultati attesi, mentre i target forniscono risultati misurabili per i tuoi carichi di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

Sviluppa obiettivi e target di costi e utilizzo per la tua organizzazione. Per un'organizzazione in crescita su AWS è importante definire e monitorare gli obiettivi ai fini dell'ottimizzazione dei costi. Questi obiettivi o indicatori chiave delle prestazioni (KPI) possono includere elementi come la percentuale della spesa on demand o l'adozione di determinati servizi ottimizzati, come le istanze AWS Graviton o i tipi di volume gp3 EBS. La definizione di obiettivi misurabili e raggiungibili ti aiuta a calcolare i miglioramenti dell'efficienza, un fattore importante per le operazioni aziendali. Gli obiettivi forniscono all'organizzazione linee guida e indicazioni sui risultati previsti.

I target forniscono i risultati specifici e misurabili da raggiungere. In breve, l'obiettivo è la direzione in cui desideri andare, mentre il target è la distanza da percorrere in quella direzione e il momento in cui l'obiettivo deve essere raggiunto, utilizzando la guida SMART, specifica, misurabile, assegnabile, realistica e tempestiva. Un esempio di obiettivo è che l'utilizzo della piattaforma aumenti in modo significativo, con solo un piccolo incremento (non lineare) dei costi. Un esempio di target è un aumento del 20% dell'utilizzo della piattaforma, con un incremento dei costi inferiore al 5%. Un altro obiettivo comune è che i carichi di lavoro devono essere più efficienti ogni sei mesi. L'obiettivo

corrispondente prevede che il costo per metrica aziendale debba diminuire del cinque per cento ogni sei mesi. Usa le metriche giuste e imposta i KPI calcolati per l'organizzazione. Puoi iniziare con i KPI di base e cambiare successivamente in base alle esigenze aziendali.

Un obiettivo per l'ottimizzazione dei costi è l'incremento dell'efficienza del carico di lavoro, ossia la riduzione del costo per ogni risultato aziendale del carico di lavoro nel corso del tempo. Implementa questo obiettivo per tutti i carichi di lavoro e stabilisci un target come l'incremento dell'efficienza del 5% ogni 6-12 mesi. Nel cloud, puoi raggiungere questo target attraverso la definizione della capacità di ottimizzazione dei costi, nonché nuove versioni di servizi e funzionalità.

I target sono i benchmark quantificabili che desideri raggiungere per conseguire i tuoi obiettivi e che confrontano i tuoi risultati effettivi rispetto al target. Stabilisci i benchmark con i KPI per il costo unitario dei servizi di calcolo, come l'adozione di istanze spot, l'adozione di Graviton, i tipi di istanza più recenti e la copertura on demand, dei servizi di archiviazione, come l'adozione di EBS GP3, gli snapshot EBS obsoleti e l'archiviazione standard Amazon S3, oppure dei servizi di database, come i motori open source RDS, l'adozione di Graviton e la copertura on demand. Questi benchmark e KPI possono aiutarti a verificare che i servizi AWS vengano usati nel modo più conveniente.

La tabella seguente fornisce un elenco di metriche standard AWS di riferimento. Ogni organizzazione può avere valori target diversi per questi KPI.

Category	KPI (%)	Description
Compute	EC2 usage Coverage	EC2 instances (in cost or hours) using SP+RI+Spot compared to total (in cost or hours) of EC2 instances
Compute	Compute SP/RI utilization	Utilized SP or RI hours compared to total available SP or RI hours
Compute	EC2/Hour cost	EC2 cost divided by the number of EC2 instances running in that hour
Compute	vCPU cost	Cost per vCPU for all instances

Category	KPI (%)	Description
Compute	Latest Instance Generation	Percentage of instances on Graviton (or other modern generation instance types)
Database	RDS coverage	RDS instances (in cost or hours) using RI compared to total (in cost or hours) of RDS instances
Database	RDS utilization	Utilized RI hours compared to total available RI hours
Database	RDS uptime	RDS cost divided by the number of RDS instances running in that hour
Database	Latest Instance Generation	Percentage of instances on Graviton (or other modern instance types)
Storage	Storage utilization	Optimized storage cost (for example Glacier, deep archive, or Infrequent Access) divided by total storage cost

Category	KPI (%)	Description
Tagging	Untagged resources	Cost Explorer: 1. Filtra crediti, sconti, tasse, rimborsi, marketplace e copia l'ultimo costo mensile. 2. Seleziona Mostra solo le risorse prive di tag in Cost Explorer. 3. Dividi l'importo delle risorse prive di tag per il costo mensile.

Utilizzando questa tabella, stabilisci i valori target o benchmark che devono essere calcolati in base agli obiettivi dell'organizzazione. Per definire KPI accurati e realistici dovrai misurare determinate metriche e comprendere i risultati aziendali per il carico di lavoro. Quando valuti le metriche delle prestazioni di un'organizzazione, tieni in considerazione i vari tipi di metrica che servono a scopi diversi. Queste metriche misurano principalmente le prestazioni e l'efficienza dell'infrastruttura tecnica piuttosto che direttamente l'impatto aziendale complessivo. Ad esempio, possono tenere traccia dei tempi di risposta del server, della latenza della rete o dei tempi di attività del sistema. Queste metriche sono fondamentali per valutare in che misura l'infrastruttura supporta le operazioni tecniche dell'organizzazione. Tuttavia, non forniscono approfondimenti diretti sugli obiettivi aziendali più ampi, come la soddisfazione del cliente, la crescita dei ricavi o la quota di mercato. Per acquisire un quadro completo delle prestazioni aziendali, integra queste metriche dell'efficienza con le metriche aziendali strategiche che sono direttamente correlate ai risultati aziendali.

Ottieni una visibilità quasi in tempo reale sui KPI e sulle relative opportunità di risparmio e monitora lo stato di avanzamento nel tempo. Per iniziare la definizione e il monitoraggio degli obiettivi KPI, consigliamo di usare la dashboard dei KPI di Cloud Intelligence Dashboards (CID). Sulla base dei dati disponibili nel report di costi e utilizzo (CUR), la dashboard dei KPI fornisce una serie di KPI consigliati per l'ottimizzazione dei costi con la possibilità di definire obiettivi personalizzati e monitorare lo stato di avanzamento nel tempo.

Se disponi di un'altra soluzione per impostare e monitorare gli obiettivi KPI, assicurati che sia adottata da tutte le parti interessate nella gestione finanziaria del cloud della tua organizzazione.

Passaggi dell'implementazione

- Definisci i livelli di utilizzo previsti: per iniziare, concentrati sui livelli di utilizzo. Coinvolgi i responsabili dell'applicazione, i team di marketing e i team aziendali a livello più ampio per capire quali sono i livelli di utilizzo previsti per il carico di lavoro. Considera in che modo potrà cambiare la domanda dei clienti nel corso del tempo e se ci saranno modifiche dovute a incrementi stagionali o campagne di marketing.
- Definisci risorse e costi del carico di lavoro: con i livelli di utilizzo definiti, quantifica le modifiche delle risorse del carico di lavoro necessarie per soddisfare questi livelli di utilizzo. Potresti dover aumentare le dimensioni o il numero di risorse per un componente del carico di lavoro, aumentare il trasferimento dei dati o modificare i componenti del carico di lavoro in un servizio diverso a un livello specifico. Specifica i costi per ciascuno di questi punti e prevedine la variazione in caso di modifica dell'utilizzo.
- Definisci gli obiettivi aziendali: considera l'output delle variazioni previste in termini di utilizzo e costi, combinalo con le modifiche previste nella tecnologia o in qualsiasi programma in esecuzione e sviluppa gli obiettivi per il carico di lavoro. Gli obiettivi devono riguardare l'utilizzo e il costo, nonché la relazione tra i due. Gli obiettivi devono essere semplici, di alto livello e aiutare le persone a capire cosa si aspetta l'azienda in termini di risultati, come avere la certezza che le risorse non utilizzate rimangano al di sotto di determinati livelli di costo. Non è necessario definire gli obiettivi per ogni tipo di risorsa non utilizzato o definire i costi causati dalle perdite per gli obiettivi e i target. Assicurati che siano disponibili programmi a livello di organizzazione (ad esempio lo sviluppo di competenze come la formazione e l'istruzione), se ci sono variazioni previste dei costi senza variazioni di utilizzo.
- Definisci i target: per ciascuno degli obiettivi definiti, specifica un target misurabile. Se l'obiettivo è aumentare l'efficienza nel carico di lavoro, il target quantifica il miglioramento (generalmente espresso in risultati aziendali per dollaro speso) e il momento in cui sarà efficace. Ad esempio, potresti definire un obiettivo per ridurre al minimo gli sprechi dovuti al provisioning eccessivo. Con questo obiettivo, il target può stabilire che gli sprechi dovuti al provisioning eccessivo di calcolo nel primo livello dei carichi di lavoro di produzione non superino il 10% del costo di calcolo del livello. Inoltre, un secondo target potrebbe stabilire che gli sprechi dovuti al provisioning eccessivo di calcolo nel secondo livello dei carichi di lavoro di produzione non superino il 5% del costo di calcolo del livello.

Risorse

Documenti correlati:

- Politiche gestite da AWS per le funzioni dell'attività
- Strategia di fatturazione con account multipli di AWS
- Control access to Regioni AWS using IAM policies
- S.M.A.R.T. Goals
- How to track your cost optimization KPIs with the CID KPI Dashboard

Video correlati:

- Well-Architected Labs: obiettivi e target (Livello 100)

Esempi correlati:

- What is a unit metric?
- Selecting a unit metric to support your business
- Unit metrics in practice – lessons learned
- How unit metrics help create alignment between business functions
- Well-Architected Labs: Decommission resources (Goals and Targets)
- Well-Architected Labs: Resource Type, Size and Number (Goals and Targets)

COST02-BP03 Implementazione di una struttura di account

Implementa una struttura di account che si adatta alla tua organizzazione. In questo modo sarà possibile ripartire e gestire i costi in tutta quanta l’organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

AWS Organizations consente di creare molteplici Account AWS che possono aiutare a governare centralmente l'ambiente mentre si dimensionano i carichi di lavoro su AWS. È possibile modellare la propria gerarchia organizzativa raggruppando gli Account AWS in una struttura di unità organizzative (OU) e creando molteplici Account AWS sotto ogni OU. Per creare una struttura di account, è necessario decidere innanzitutto quale Account AWS sarà l'account di gestione. Successivamente, è possibile creare nuovi Account AWS o selezionare account esistenti come account membri in base alla struttura degli account progettata, seguendo le best practice per gli account di gestione e le best practice per gli account membri.

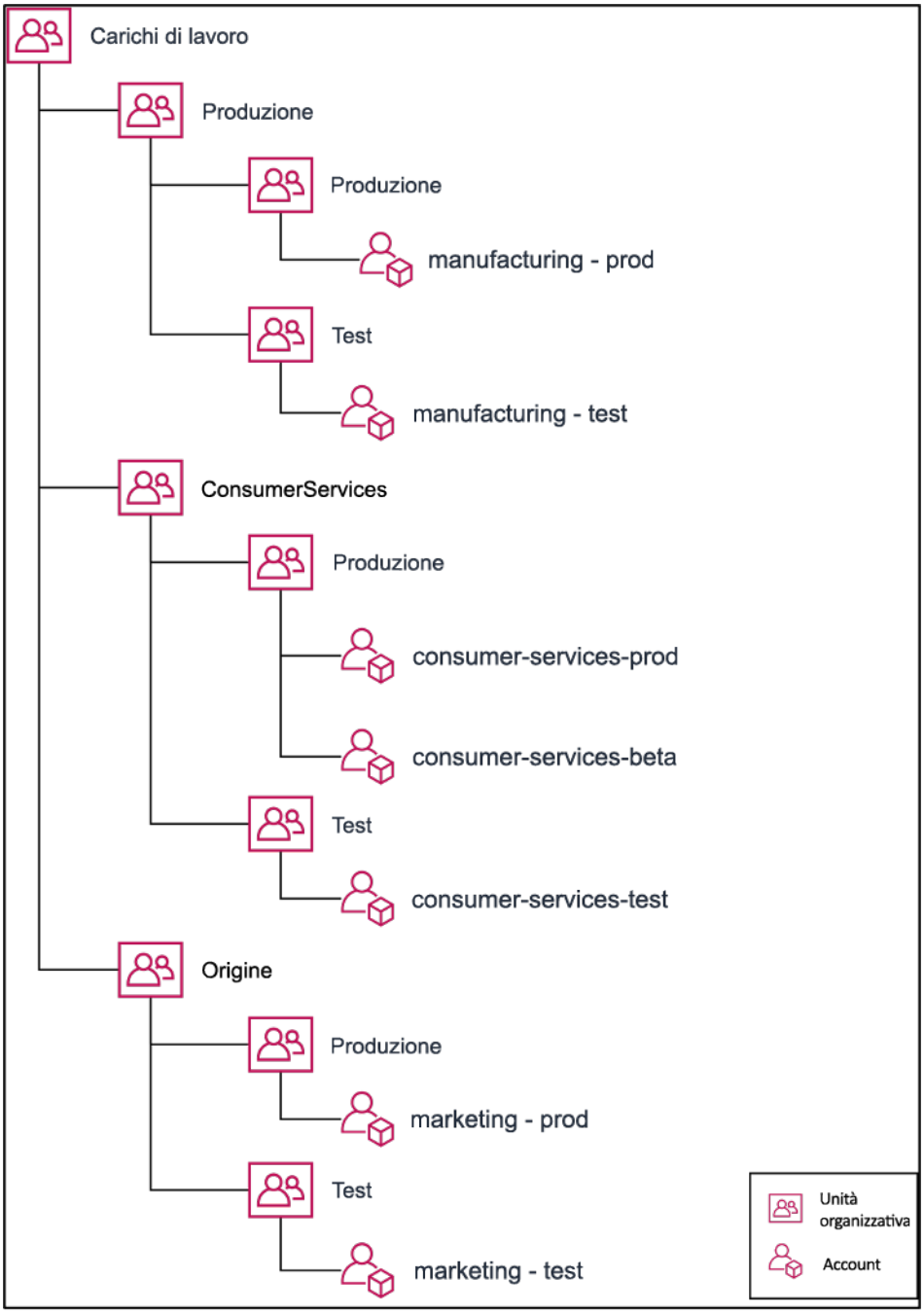
È consigliabile disporre sempre di almeno un account di gestione con un account membro collegato, indipendentemente dalle dimensioni dell'organizzazione o dall'utilizzo. Tutte le risorse del carico di lavoro dovrebbero risiedere solo all'interno degli account membri e nessuna risorsa dovrebbe essere creata all'interno dell'account di gestione. Non esiste una risposta giusta o sbagliata in merito al numero di Account AWS che bisognerebbe creare. Valuta i tuoi modelli operativi e di costo attuali e futuri per assicurarti che la struttura dei tuoi Account AWS rispecchi quella della tua organizzazione. Alcune aziende creano molteplici Account AWS per motivi aziendali, ad esempio:

- È richiesto l'isolamento amministrativo o fiscale e di fatturazione tra unità aziendali o centri di costo o carichi di lavoro specifici.
- Le restrizioni dei servizi AWS sono impostate in modo che risultino specifiche per determinati carichi di lavoro.
- Esiste un requisito per l'isolamento e la separazione tra carichi di lavoro e risorse.

All'interno di AWS Organizations, la fatturazione consolidata crea il costrutto tra uno o più account membri e l'account di gestione. Gli account membri consentono di isolare e distinguere i costi e l'utilizzo per gruppi. Una pratica comune è quella di avere account membri separati per ciascuna unità aziendale (come finanza, marketing e vendite), per il ciclo di vita di ciascun ambiente (come sviluppo, test e produzione) o per ciascun carico di lavoro (carico di lavoro a, b e c) e poi aggregare questi account membri tramite la fatturazione consolidata.

La fatturazione consolidata consente di accorpare i pagamenti di più Account AWS membri sotto un unico account di gestione e, al tempo stesso, di fornire comunque visibilità all'attività di ciascun account membro. Il fatto che i costi e l'utilizzo vengono aggregati nell'account di gestione consente di massimizzare gli sconti per volume di servizio e di massimizzare l'utilizzo degli sconti a fronte di impegni (Savings Plans e istanze riservate) per ottenere gli sconti più elevati.

Il diagramma seguente mostra come è possibile utilizzare AWS Organizations con le unità organizzative (OU) per raggruppare più account e come inserire molteplici Account AWS sotto ciascuna OU. Si consiglia di utilizzare le OU per diversi casi d'uso e carichi di lavoro che forniscono modelli per l'organizzazione degli account.



Esempio di raggruppamento di molteplici Account AWS all'interno di unità organizzative.

AWS Control Tower può impostare e configurare rapidamente più account AWS, garantendo una governance in linea con i requisiti dell'organizzazione.

Passaggi dell'implementazione

- Definisci i requisiti di separazione: i requisiti di separazione sono una combinazione di più fattori, tra cui sicurezza, affidabilità e costrutti finanziari. Analizza ciascun fattore in ordine e specifica se

il carico di lavoro o l'ambiente del carico di lavoro deve essere separato da altri carichi di lavoro. La sicurezza riguarda il rispetto dei requisiti di accesso e di dati. L'affidabilità riguarda la gestione dei limiti, in modo che gli ambienti e i carichi di lavoro non influiscano gli uni sugli altri. Esamina periodicamente i pilastri della sicurezza e dell'affidabilità del Canone di architettura e segui le best practice messe a disposizione. I costrutti finanziari creano una rigida separazione finanziaria (centri di costo diversi, proprietà e responsabilità dei carichi di lavoro). Esempi comuni di separazione sono i carichi di lavoro di produzione e test eseguiti in account separati o l'utilizzo di un account separato in modo che i dati di fatturazione possano essere forniti ai singoli settori o reparti aziendali dell'organizzazione o alle terze parti proprietarie dell'account.

- Definisci i requisiti di raggruppamento: i requisiti per il raggruppamento non sostituiscono i requisiti di separazione, ma vengono utilizzati a supporto della gestione. Raggruppa ambienti o carichi di lavoro simili che non richiedono separazione. Un esempio è costituito dal raggruppamento di più ambienti di test o sviluppo associati a uno o più carichi di lavoro.
- Definisci la struttura dell'account: utilizzando queste separazioni e questi raggruppamenti, specifica un account per ogni gruppo e mantieni i requisiti di separazione. Questi account sono i tuoi account membri o collegati. Raggruppando questi account membri in un unico account di gestione o di pagamento, puoi combinarne l'utilizzo, ottenendo maggiori sconti per i volumi e una singola fattura per tutti gli account. È possibile separare i dati di fatturazione e fornire a ciascun account membro una visualizzazione individuale su di essi. Se un account membro non deve avere i dati di utilizzo o di fatturazione visibili a qualsiasi altro account, oppure se è necessaria una fattura separata da parte di AWS, definisci più account di gestione/di pagamento. In questo caso, ogni account membro ha il proprio account di gestione o di pagamento. Le risorse devono sempre essere collocate negli account membri o collegati. Gli account di gestione/di pagamento devono essere utilizzati solo per la gestione.

Risorse

Documenti correlati:

- Utilizzo dei tag di allocazione dei costi
- Politiche gestite da AWS per le funzioni dell'attività
- Strategia di fatturazione con account multipli di AWS
- Controllo dell'accesso a Regioni AWS tramite le politiche IAM
- AWS Control Tower
- AWS Organizations

- Best practice per account di gestione e account membri
- Organizzazione dell'ambiente AWS con l'utilizzo di account multipli
- Attivazione delle istanze riservate condivise e degli sconti dei Savings Plans
- Fatturazione consolidata
- Fatturazione consolidata

Esempi correlati:

- Divisione del Report costi e utilizzo (CUR) e condivisione dell'accesso

Video correlati:

- Presentazione di AWS Organizations
- Impostazione di un ambiente AWS multi-account che utilizzi le best practice di AWS Organizations

Esempi correlati:

- Well-Architected Labs: creazione di un'organizzazione AWS (Livello 100)
- Divisione del AWS Cost and Usage Report e condivisione dell'accesso
- Definizione di una strategia AWS multi-account per le aziende di telecomunicazioni
- Best practice per l'ottimizzazione di Account AWS
- Best practice per le unità organizzative con AWS Organizations

COST02-BP04 Implementazione di gruppi e ruoli

Implementa gruppi e ruoli che si allineino alle tue policy e controlla chi può creare, modificare o ritirare istanze e risorse in ogni gruppo. Ad esempio, implementa gruppi di sviluppo, test e produzione. Questo si applica ai servizi AWS e a soluzioni di terze parti.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

I ruoli e i gruppi di utenti sono elementi costitutivi fondamentali nella progettazione e implementazione di sistemi sicuri ed efficienti. I ruoli e i gruppi aiutano le organizzazioni a trovare il giusto equilibrio

a livello di controllo dei requisiti di flessibilità e produttività, supportando in definitiva gli obiettivi organizzativi e le esigenze degli utenti. Come consigliato nella sezione Gestione di identità e accessi del Pilastro della sicurezza del Framework AWS Well-Architected, è necessaria una solida gestione delle identità e delle autorizzazioni per fornire l'accesso alle risorse giuste alle persone idonee nelle condizioni adatte. Gli utenti disporranno solo del livello di accesso necessario per completare le proprie attività. Ciò riduce al minimo il rischio associato all'accesso non autorizzato o all'uso improprio.

Dopo avere sviluppato le policy, è possibile creare gruppi logici e ruoli degli utenti all'interno dell'organizzazione. Ciò consente di assegnare le autorizzazioni, controllare l'utilizzo e semplificare l'implementazione di affidabili meccanismi di controllo degli accessi, impedendo l'accesso non autorizzato alle informazioni sensibili. Inizia con i raggruppamenti di persone di alto livello. Generalmente questi corrispondono alle unità organizzative e ai ruoli lavorativi (ad esempio, amministratore di sistema nel reparto IT, controllore finanziario o business analyst). I gruppi classificano le persone che eseguono attività simili e necessitano di un accesso simile. I ruoli definiscono che cosa un gruppo deve fare. È più facile gestire le autorizzazioni per gruppi e ruoli che per i singoli utenti. I ruoli e i gruppi assegnano le autorizzazioni in modo coerente e sistematico a tutti gli utenti, evitando errori e incongruenze.

Quando il ruolo di un utente cambia, gli amministratori possono modificare l'accesso a livello di ruolo o di gruppo, anziché riconfigurare i singoli account utente. Ad esempio, un amministratore di sistema nel reparto IT deve disporre di un accesso che permetta di creare tutte le risorse, mentre un membro del team di analisi ha la necessità di creare soltanto risorse di analisi.

Passaggi dell'implementazione

- Implementazione dei gruppi: utilizzando i gruppi di utenti definiti nelle policy dell'organizzazione, implementa i gruppi corrispondenti, se necessario. Per le best practice su utenti, gruppi e autenticazione, consulta il Pilastro della sicurezza del Framework AWS Well-Architected.
- Implementazione di ruoli e policy: utilizzando le operazioni definite nelle policy dell'organizzazione, crea le policy di accesso e i ruoli necessari. Per le best practice su ruoli e policy, consulta il Pilastro della sicurezza del Framework AWS Well-Architected.

Risorse

Documenti correlati:

- Policy gestite da AWS per le funzioni lavorative

- Strategia di fatturazione con account multipli di AWS
- Pilastro della sicurezza del Framework AWS Well-Architected
- AWS Identity and Access Management (IAM)
- Policy AWS Identity and Access Management

Video correlati:

- AWS Identity and Access Management (IAM)

Esempi correlati:

- Well-Architected Labs: Identità e accesso base
- Easier way to control access to Regioni AWS using IAM policies
- Starting your Cloud Financial Management journey: Cloud cost operations

COST02-BP05 Implementazione dei controlli di costo

Implementa controlli basati sulle policy dell'organizzazione e gruppi e ruoli definiti. Questi garantiscono che i costi siano sostenuti solo in base ai requisiti dell'organizzazione come, ad esempio, il controllo dell'accesso alle regioni o ai tipi di risorse.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

Un primo passo comune per implementare i controlli dei costi consiste nell'impostare delle notifiche quando si verificano eventi di costi o utilizzo al di fuori delle policy. In questo modo è possibile agire rapidamente e verificare se è necessaria un'azione correttiva, senza limitare o influire negativamente sui carichi di lavoro o sulle nuove attività. Dopo avere appreso i limiti del carico di lavoro e dell'ambiente, puoi applicare la governance. Budget AWS consente di impostare notifiche e di definire budget mensili per i costi, l'utilizzo e gli sconti a fronte di impegni di AWS (Savings Plans e istanze riservate). È possibile creare budget a livello di costo aggregato (ad esempio, tutti i costi) o a un livello più granulare, includendo solo dimensioni specifiche come account membri, servizi, tag o zone di disponibilità.

Una volta configurati i limiti di budget con Budget AWS, utilizza AWS Cost Anomaly Detection per ridurre i costi inaspettati. AWS Cost Anomaly Detection è un servizio di gestione dei costi che utilizza

il machine learning per monitorare costantemente i costi e l'utilizzo con lo scopo di individuare le spese anomale. Aiuta a individuare le spese anomale e le cause principali in modo da garantire un intervento tempestivo. Per prima cosa, crea un monitor dei costi in AWS Cost Anomaly Detection, quindi scegli le tue preferenze relativamente agli avvisi impostando una soglia in dollari (ad esempio, un avviso sulle spese anomale con impatto superiore a 1.000 dollari). Una volta ricevuti gli avvisi, puoi analizzare la causa alla base dell'anomalia e l'impatto sui costi. Puoi inoltre monitorare ed eseguire la tua analisi delle anomalie in AWS Cost Explorer.

Imponi le policy di governance in AWS attraverso AWS Identity and Access Management e le Policy di controllo dei servizi (Service Control Policies, SCP) di AWS Organizations. IAM permette di gestire in modo sicuro l'accesso ai servizi e alle risorse di AWS. Utilizzando IAM, puoi controllare chi può creare e gestire le risorse di AWS, il tipo di risorse che possono essere create e dove possono essere create. In questo modo riduci al minimo la possibilità che vengano create risorse al di fuori della policy definita. Utilizza i ruoli e i gruppi creati in precedenza e assegna le policy IAM per garantire l'utilizzo corretto. Le SCP offrono il controllo centralizzato sul numero massimo di autorizzazioni disponibili per tutti gli account nella tua organizzazione, assicurando che i tuoi account rimangano entro le linee guida di controllo degli accessi. Le SCP sono disponibili soltanto in un'organizzazione con tutte le funzionalità abilitate e possono essere configurate in modo da rifiutare o consentire operazioni agli account membri per impostazione predefinita. Per ulteriori dettagli sull'implementazione della gestione degli accessi, consulta il whitepaper sul principio della sicurezza del canone di architettura.

La governance può essere implementata anche tramite la gestione delle Service Quotas di AWS. Assicurandoti che le Service Quotas siano impostate con spese minime e siano gestite in modo accurato, puoi ridurre al minimo la creazione di risorse che non rientrano nei requisiti della tua organizzazione. Per ottenere questo risultato, devi comprendere la velocità con cui i tuoi requisiti possono cambiare, valutare i progetti in corso (sia la creazione sia la disattivazione di risorse) e considerare la velocità con cui è possibile implementare le modifiche alle quote. Le Service Quotas possono essere utilizzate per aumentare le quote all'occorrenza.

Passaggi dell'implementazione

- Implementa le notifiche delle spese: utilizzando le policy dell'organizzazione definite, crea dei Budget AWS per inviare notifiche quando la spesa ricade al di fuori delle policy. Configura più budget dei costi, uno per ogni account, in modo da ricevere informazioni sulla spesa complessiva del conto. Quindi configura budget di costo aggiuntivi all'interno di ciascun account per unità più piccole al suo interno. Queste unità variano a seconda della struttura dell'account. Alcuni esempi comuni sono Regioni AWS, carichi di lavoro (tramite i tag) o servizi AWS. Configura un elenco

di distribuzione e-mail come destinatario per le notifiche e non un account e-mail di una singola persona. È possibile configurare un budget effettivo per quando un importo viene superato oppure utilizzare un budget previsto per la notifica dell'utilizzo previsto. Si possono anche preconfigurare AWS Budget Actions (operazioni di budget) che possono applicare specifiche policy IAM o SCP o arrestare delle istanze Amazon EC2 o Amazon RDS definite. Le operazioni di budget possono essere eseguite automaticamente o richiedere l'approvazione del flusso di lavoro.

- Implementa le notifiche sulle spese anomale: utilizza AWS Cost Anomaly Detection per ridurre i costi inattesi nella tua organizzazione e analizzare le cause di potenziali spese anomale. Una volta creato il sistema di monitoraggio dei costi per identificare le spese insolite alla granularità specificata e aver configurato le notifiche in AWS Cost Anomaly Detection, viene inviato un avviso quando sono rilevate spese insolite. Questo ti permetterà di analizzare le cause alla base dell'anomalia e di valutarne l'impatto sui costi. Utilizza le categorie di costo AWS durante la configurazione di AWS Cost Anomaly Detection per identificare il team di progetto o il team della business unit che può analizzare la causa principale del costo imprevisto e intraprendere tempestivamente le azioni necessarie.
- Implementa il controllo dell'utilizzo utilizzando le policy dell'organizzazione definite, implementa policy e ruoli IAM per specificare quali azioni possono o non possono eseguire gli utenti. In una policy AWS possono essere incluse più policy organizzative. Nello stesso modo in sono state definite le policy, occorre iniziare in modo generale e quindi applicare controlli più dettagliati a ogni fase. Anche le restrizioni dei servizi sono un controllo efficace sull'utilizzo. Implementa le restrizioni dei servizi corrette su tutti gli account.

Risorse

Documenti correlati:

- Politiche gestite da AWS per le funzioni dell'attività
- Strategia di fatturazione con account multipli di AWS
- Controllo dell'accesso a Regioni AWS tramite le politiche IAM
- Budget AWS
- AWS Cost Anomaly Detection
- Controlla i tuoi costi di AWS

Video correlati:

- Come faccio a utilizzare Budget AWS per tenere traccia delle mie spese e del mio utilizzo

Esempi correlati:

- Policy di gestione degli accessi IAM di esempio
- Policy di controllo dei servizi di esempio
- AWS Budget Actions
- Crea una policy IAM per controllare l'accesso alle risorse Amazon EC2 utilizzando i tag
- Limit l'accesso dell'identità IAM a specifiche risorse Amazon EC2
- Crea una policy IAM per limitare l'uso di Amazon EC2 a famiglie di macchine selezionate
- Well-Architected Labs: utilizzo di costi e governance (Livello 100)
- Well-Architected Labs: utilizzo di costi e governance (Livello 200)
- Integrazioni Slack per Cost Anomaly Detection utilizzando AWS Chatbot

COST02-BP06 Monitoraggio del ciclo di vita del progetto

Rileva, misura e controlla il ciclo di vita di progetti, team e ambienti per evitare di usare risorse non necessarie e pagare per esse.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

Monitorando efficacemente il ciclo di vita del progetto, le organizzazioni possono ottimizzare il controllo dei costi attraverso una migliore pianificazione, gestione e ottimizzazione delle risorse. Gli approfondimenti acquisiti attraverso il monitoraggio sono preziose per la formulazione di decisioni informate che contribuiscono alla competitività a livello di costi e al successo complessivo del progetto.

Il monitoraggio dell'intero ciclo di vita del carico di lavoro consente di capire quando i carichi di lavoro o i suoi componenti non sono più necessari. I carichi di lavoro e i componenti esistenti possono sembrare in uso, ma quando AWS rilascia nuovi servizi o nuove funzionalità, è possibile che vengano disattivati o adottati. Controlla le fasi precedenti dei carichi di lavoro. Quando un carico di lavoro arriva in produzione, gli ambienti precedenti possono essere disattivati o notevolmente ridotti in termini di capacità fino a quando non sono nuovamente necessari.

Puoi applicare i tag alle risorse con un intervallo di tempo o un promemoria per aggiungere l'ora in cui il carico di lavoro è stato esaminato. Ad esempio, se sono trascorsi mesi dall'ultima volta che l'ambiente di sviluppo è stato esaminato, potrebbe essere il momento giusto per esaminarlo

nuovamente per verificare se è possibile adottare nuovi servizi o se l'ambiente è in uso. Puoi raggruppare e applicare i tag alle tue applicazioni con myApplications su AWS per gestire e controllare i metadati quali criticità, ambiente, ultima revisione e centro di costo. Puoi tenere traccia del ciclo di vita del carico di lavoro e monitorare e gestire i costi, lo stato di integrità, il livello di sicurezza e le prestazioni delle applicazioni.

AWS offre diversi servizi di gestione e governance utilizzabili per il monitoraggio del ciclo di vita delle entità. Puoi usare AWS Config o AWS Systems Manager per fornire un inventario dettagliato delle risorse e della configurazione AWS. Ti consigliamo di integrare questi servizi con i sistemi di gestione di progetti o asset esistenti per tenere traccia dei progetti attivi e dei prodotti all'interno della tua organizzazione. La combinazione del tuo sistema attuale con l'ampia gamma di eventi e parametri forniti da AWS ti consentirà di ottenere una panoramica degli eventi del ciclo di vita significativi e di gestire le risorse in modo proattivo per ridurre i costi non necessari.

Analogamente alla gestione del ciclo di vita dell'applicazione (ALM), il monitoraggio del ciclo di vita del progetto dovrebbe coinvolgere più processi, strumenti e team che interagiscono tra loro, ad esempio progettazione e sviluppo, test, produzione, supporto e ridondanza del carico di lavoro.

Monitorando attentamente ogni fase del ciclo di vita di un progetto, le organizzazioni ottengono informazioni cruciali e un maggiore controllo, semplificando la pianificazione, l'implementazione e la riuscita del progetto. Questa attenta supervisione verifica che i progetti non solo soddisfino gli standard di qualità, ma vengano consegnati in tempo e nel rispetto del budget, promuovendo l'efficienza complessiva dei costi.

Per ulteriori dettagli sull'implementazione del monitoraggio del ciclo di vita delle entità, consulta il whitepaper Principio dell'eccellenza operativa di AWS Well-Architected.

Passaggi dell'implementazione

- Definisci un processo di monitoraggio del ciclo di vita del progetto: il team del Centro di eccellenza del Cloud (CCoE) deve stabilire un processo di monitoraggio del ciclo di vita del progetto. Stabilisci un approccio strutturato e sistematico per il monitoraggio dei carichi di lavoro al fine di migliorare il controllo, la visibilità e le prestazioni dei progetti. Rendi il processo di monitoraggio trasparente, collaborativo e incentrato sul miglioramento continuo per massimizzarne l'efficacia e il valore.
- Esegui le revisioni del carico di lavoro: come definito dalle policy dell'organizzazione, imposta una cadenza regolare per controllare i progetti esistenti ed eseguire le revisioni del carico di lavoro. L'impegno speso per il controllo deve essere proporzionale al rischio, al valore o al costo approssimativo per l'organizzazione. Le aree chiave da includere nell'audit sono il rischio di incidente o interruzione per l'organizzazione, il valore o contributo all'organizzazione (misurato

in fatturato o reputazione del marchio), il costo del carico di lavoro (misurato come costo totale delle risorse e costi operativi) e l'utilizzo del carico di lavoro (misurato in numero di risultati dell'organizzazione per unità di tempo). Se queste aree cambiano durante il ciclo di vita, sono necessarie modifiche al carico di lavoro, ad esempio la disattivazione completa o parziale.

Risorse

Documenti correlati:

- Guidance for Tagging on AWS
- What is ALM (Application Lifecycle Management)?
- AWS managed policies for job functions

Esempi correlati:

- Control access to Regioni AWS using IAM policies

Strumenti correlati

- AWS Config
- AWS Systems Manager
- Budget AWS
- AWS Organizations
- AWS CloudFormation

COST 3. In che modo monitori i costi e l'utilizzo?

Stabilisci policy e procedure per monitorare e allocare i costi in modo appropriato. Ciò ti permette di misurare e migliorare l'efficienza in termini di costi del carico di lavoro.

Best practice

- COST03-BP01 Configurazione di fonti di informazione dettagliate
- COST03-BP02 Aggiunta di informazioni sull'organizzazione a costi e utilizzo
- COST03-BP03 Identificazione delle categorie di attribuzione dei costi
- COST03-BP04 Definizione dei parametri dell'organizzazione

- COST03-BP05 Configurazione degli strumenti di fatturazione e di gestione dei costi
- COST03-BP06 Allocazione dei costi in base alle metriche del carico di lavoro

COST03-BP01 Configurazione di fonti di informazione dettagliate

Imposta gli strumenti di gestione e report dei costi per ottenere una migliore analisi e trasparenza dei dati sui costi e sull'utilizzo. Configura il carico di lavoro per creare voci di log che facilitino il monitoraggio e la segregazione dei costi e dell'utilizzo.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Informazioni dettagliate sulla fatturazione, come la granularità oraria negli strumenti di gestione dei costi, consentono alle organizzazioni di tenere traccia dei propri consumi con ulteriori dettagli e di aiutarle a identificare alcuni dei motivi di aumento dei costi. Queste origini dati forniscono la visualizzazione più accurata dei costi e dell'utilizzo dell'intera organizzazione.

Puoi usare Esportazioni di dati AWS per creare esportazioni di AWS Cost and Usage Report (CUR) 2.0. Si tratta del nuovo modo consigliato per ricevere dati dettagliati su costi e utilizzo da AWS. Fornisce una granularità di utilizzo giornaliera o oraria, tariffe, costi e attributi di utilizzo per tutti i servizi AWS a pagamento (le stesse informazioni del CUR), oltre ad alcuni miglioramenti. Nel CUR sono inclusi tutti gli aspetti possibili, compresi tag, posizione, attributi delle risorse e ID account.

Esistono tre tipi di esportazione in base a ciò che desideri creare: l'esportazione di dati standard, l'esportazione in una dashboard costi e utilizzo con l'integrazione di Amazon QuickSight oppure l'esportazione di dati legacy.

- Esportazione di dati standard: l'esportazione personalizzata di una tabella che viene distribuita in Amazon S3 su base ricorrente.
- Esportazione in una dashboard costi e utilizzo: l'esportazione e l'integrazione di Amazon QuickSight per implementare una dashboard costi e utilizzo predefinita.
- Esportazione di dati legacy: l'esportazione dell'AWS Cost and Usage Report (CUR) legacy.

È possibile creare esportazioni di dati con le seguenti personalizzazioni:

- Inclusione degli ID risorsa
- Dati di allocazione dei costi suddivisi

- Granularità oraria
- Versioning
- Tipo di compressione e formato del file

Per i carichi di lavoro che eseguono container su Amazon ECS o Amazon EKS, abilita i dati di allocazione dei costi suddivisi in modo da poter allocare i costi dei container a singole business unit e team, in base al modo in cui i carichi di lavoro dei container consumano le risorse di calcolo e memoria condivise. I dati di allocazione dei costi suddivisi introducono in AWS Cost and Usage Report i dati sui costi e sull'utilizzo per le nuove risorse a livello di container. I dati di allocazione dei costi suddivisi vengono calcolati determinando il costo di singoli servizi e attività ECS in esecuzione sul cluster.

La dashboard costi e utilizzo esporta la tabella dei costi e dell'utilizzo in un bucket S3 su base ricorrente e implementa una dashboard costi e utilizzo predefinita in Amazon QuickSight. Utilizza questa opzione se desideri implementare rapidamente una dashboard dei dati su costi e utilizzo senza funzionalità di personalizzazione.

È comunque possibile esportare il report CUR in modalità legacy per integrare altri servizi di elaborazione, ad esempio AWS Glue, per preparare i dati per l'analisi ed eseguire l'analisi dei dati con Amazon Athena utilizzando SQL per le query sui dati.

Passaggi dell'implementazione

- Crea esportazioni di dati: crea esportazioni personalizzate con i dati che desideri e controlla lo schema delle esportazioni. Crea le esportazioni dei dati di fatturazione e gestione dei costi utilizzando SQL di base e visualizza i dati di fatturazione e gestione dei costi integrandoli con Amazon QuickSight. Puoi anche esportare i dati in modalità standard per analizzarli con altri strumenti di elaborazione, come Amazon Athena.
- Configura il report su costi e utilizzo: utilizzando la console di fatturazione, configura almeno un rapporto su costi e utilizzo. Configura un report con granularità oraria che include tutti gli identificatori e gli ID risorsa. Puoi anche creare altri report con granularità diverse per fornire informazioni di riepilogo di livello superiore.
- Configura la granularità oraria in Cost Explorer: per accedere ai dati su costi e utilizzo con granularità oraria negli ultimi 14 giorni, valuta la possibilità di abilitare i dati a livello di ora e risorsa nella console di fatturazione.
- Configura la registrazione dei log da parte delle applicazioni: verifica che ogni applicazione registri ogni risultato aziendale raggiunto in modo che possa essere monitorato e misurato. Assicurati

che la granularità di questi dati sia almeno oraria, affinché possa essere abbinata ai dati relativi a costi e utilizzo. Per maggiori dettagli sulla registrazione e il monitoraggio, consulta il pilastro dell'eccellenza operativa Well-Architected.

Risorse

Documenti correlati:

- Esportazioni di dati AWS
- AWS Glue
- Amazon QuickSight
- AWS Cost Management Pricing
- Tagging AWS resources
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Report
- Well-Architected Operational Excellence Pillar

Esempi correlati:

- AWS Account Setup
- Data Exports for AWS Billing and Cost Management
- AWS Cost Explorer Common Use Cases

COST03-BP02 Aggiunta di informazioni sull'organizzazione a costi e utilizzo

Definisci uno scherma per l'applicazione di tag in base alla tua organizzazione, agli attributi del carico di lavoro e alle categorie di allocazione dei costi, in modo da poter filtrare e cercare le risorse o monitorare costi e utilizzo negli strumenti di gestione dei costi. Implementa un'applicazione di tag consistente in tutte le risorse possibili per scopo, team, ambiente o altri criteri pertinenti alla tua azienda.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

Implementa l'applicazione di tag in AWS per aggiungere informazioni sull'organizzazione alle risorse, che verranno, quindi, aggiunte alle informazioni su costi e utilizzo. Un tag è una coppia chiave-

valore: la chiave è definita e deve essere univoca all'interno dell'organizzazione, mentre il valore è univoco per un gruppo di risorse. Ad esempio, una coppia chiave-valore può essere costituita da `ambiente` (chiave) e `produzione` (valore). Tutte le risorse nell'ambiente di produzione avranno questa coppia chiave-valore. L'applicazione di tag consente di categorizzare e monitorare i costi con informazioni significative e rilevanti sull'organizzazione. Puoi applicare tag che rappresentano categorie dell'organizzazione (ad esempio, centri di costo, nomi di applicazioni, progetti o proprietari) e identificano carichi di lavoro e rispettive funzionalità (come test o produzione) per attribuire i costi e l'utilizzo all'interno dell'organizzazione.

Quando applichi i tag alle tue risorse AWS (come le istanze Amazon Elastic Compute Cloud o i bucket Amazon Simple Storage Service) e li attivi, AWS aggiunge queste informazioni ai report su costi e utilizzo. Puoi creare report e condurre analisi su risorse con tag e senza tag per incrementare la conformità con le politiche di gestione dei costi interne e garantire un'attribuzione accurata.

La creazione e l'implementazione di uno standard per l'applicazione di tag AWS tra gli account dell'organizzazione ti consente di gestire e amministrare gli ambienti AWS in modo coerente e uniforme. Usa le politiche sui tag in AWS Organizations per definire regole su come i tag possono essere applicati alle risorse AWS nei tuoi account in AWS Organizations. Le policy di tag consentono di adottare con facilità un approccio standardizzato per l'applicazione di tag alle risorse AWS.

AWS Tag Editor consente di aggiungere, eliminare e gestire tag di più risorse. Con Tag Editor cerchi le risorse a cui applicare i tag, quindi gestisci i tag per le risorse nei risultati della ricerca.

AWS Cost Categories consente di assegnare ai tuoi costi significati per l'organizzazione, senza necessità di applicare tag alle risorse. Puoi mappare le informazioni su costi e utilizzo attribuendole a strutture organizzative interne univoche. Puoi definire regole di categoria per mappare e categorizzare i costi utilizzando le dimensioni di fatturazione, ad esempio account e tag. Questo offre un altro livello di funzionalità di gestione oltre all'applicazione di tag. Puoi anche mappare account e tag specifici attribuendoli a più progetti.

Passaggi dell'implementazione

- Definisci uno schema per l'applicazione di tag: riunisci tutte le parti interessate di tutta l'azienda per definire uno schema. Questo generalmente include i ruoli tecnici, finanziari e di gestione. Definisci un elenco di tag che tutte le risorse devono avere, nonché un elenco di tag che le risorse dovrebbero avere. Verifica che i nomi e i valori dei tag siano coerenti all'interno dell'organizzazione.
- Risorse di tag: utilizzando le categorie di attribuzione dei costi definite, posiziona i tag su tutte le risorse nei carichi di lavoro in base alle categorie. Utilizza strumenti come l'interfaccia a riga di comando (CLI), Tag Editor o AWS Systems Manager per aumentare l'efficienza.

- Implementa AWS Cost Categories: puoi creare delle categorie di costo senza implementare l'applicazione dei tag. Cost Categories utilizza le dimensioni di costo e utilizzo esistenti. Crea regole di categoria dallo schema e implementale in Cost Categories.
- Applicazione automatica di tag: per verificare di mantenere elevati livelli di applicazione di tag tra tutte le risorse, automatizza l'applicazione di tag in modo che le risorse siano contrassegnate automaticamente al momento della creazione. Usa i servizi come AWS CloudFormation per verificare che alle risorse vengano applicati i tag al momento della creazione. Puoi anche creare una soluzione personalizzata per applicare i tag in automatico tramite le funzioni Lambda o usare un microservizio che scansioni periodicamente il carico di lavoro e rimuova le risorse non contrassegnate, l'ideale per ambienti di test e sviluppo.
- Monitora ed elabora report sull'applicazione di tag: per verificare di mantenere elevati livelli di applicazione di tag nella tua organizzazione, segnala e monitora i tag tra i tuoi carichi di lavoro. Puoi utilizzare AWS Cost Explorer per visualizzare il costo delle risorse con tag e senza tag oppure utilizzare servizi come Tag Editor. Verifica regolarmente il numero di risorse senza tag e aggiungi i tag fino a raggiungere il livello desiderato di applicazione di tag.

Risorse

Documenti correlati:

- Best practice per l'applicazione di tag
- Applicazione di tag alle risorse AWS CloudFormation
- AWS Cost Categories
- Applicazione di tag alle risorse AWS
- Analisi dei costi con Budget AWS
- Analisi dei costi con Cost Explorer
- Gestione del report su costi e utilizzo di AWS

Video correlati:

- Come posso applicare tag alle mie risorse AWS per dividere la mia fattura per centro di costo o progetto
- Applicazione di tag alle risorse AWS

Esempi correlati:

- Applica automaticamente tag alle nuove risorse AWS in base all'identità o al ruolo

COST03-BP03 Identificazione delle categorie di attribuzione dei costi

Identifica le categorie dell'organizzazione, come unità aziendali, reparti o progetti, che potrebbero essere utilizzate per allocare i costi alle entità responsabili dei consumi interni. Utilizza queste categorie per rafforzare la responsabilità della spesa, creare consapevolezza dei costi e promuovere comportamenti di consumo efficaci.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

Il processo di categorizzazione dei costi è fondamentale nella definizione del budget, nella contabilità, nella rendicontazione finanziaria, nel processo decisionale, nell'analisi comparativa e nella gestione dei progetti. La categorizzazione e la classificazione delle spese consentono ai team di comprendere meglio i tipi di costi che dovranno sostenere durante il loro percorso verso il cloud, aiutandoli a prendere decisioni informate e a gestire i budget in modo efficace.

La responsabilità della spesa cloud costituisce un forte incentivo per una gestione disciplinata della domanda e dei costi. Il risultato è un notevole risparmio sui costi del cloud per le organizzazioni che destinano la maggior parte della loro spesa per il cloud a unità aziendali o team che utilizzano il cloud. Inoltre, l'allocazione della spesa per il cloud aiuta le organizzazioni ad adottare un numero maggiore di best practice di governance centralizzata del cloud.

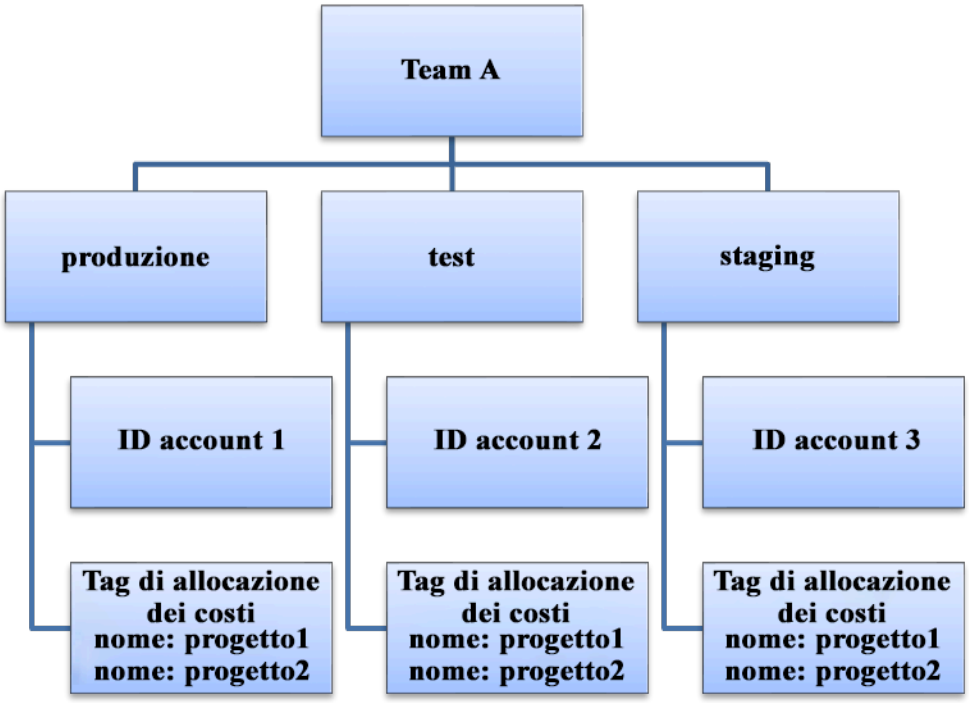
Collabora con il tuo team finanziario e altre parti interessate per comprendere i requisiti di allocazione dei costi all'interno della tua organizzazione durante le riunioni organizzate con periodicità regolare. I costi del carico di lavoro devono essere allocati per tutto il ciclo di vita, inclusi sviluppo, test, produzione e disattivazione. Comprendi in che modo i costi sostenuti per formazione, sviluppo del personale e creazione di idee sono attribuiti all'interno dell'organizzazione. Questo può essere utile per assegnare correttamente gli account utilizzati a questo scopo ai budget di formazione e sviluppo anziché ai budget di costi IT generici.

Dopo aver definito le categorie di attribuzione dei costi con le parti interessate all'interno dell'organizzazione, utilizza Categorie di costo AWS per raggruppare le informazioni sui costi e sull'utilizzo in categorie significative nel Cloud AWS, ad esempio costi per progetti specifici o Account AWS per reparti o unità aziendali. Puoi creare categorie personalizzate e mappare le informazioni su costi e utilizzo in queste categorie in base a regole che definisci usando componenti diversi come account, tag, servizio o tipo di addebito. Una volta impostate le categorie di costi, è possibile

visualizzare le informazioni su costi e utilizzo consentendo così alla tua organizzazione di prendere decisioni di acquisto e strategiche migliori. Tali categorie sono visibili in AWS Cost Explorer, Budget AWS e anche in AWS Cost and Usage Report.

Ad esempio, crea categorie di costo per le tue unità aziendali (Team DevOps) e in ogni categoria crea più regole (regole per ogni sottocategoria) con più componenti (Account AWS, tag di allocazione dei costi, servizi o tipo di addebito) in base ai raggruppamenti da te definiti. Con le categorie di costo puoi organizzare i tuoi costi con un motore basato su regole. Le regole che configuri organizzano i costi in categorie. All'interno di queste regole, puoi filtrare utilizzando più aspetti o componenti per ciascuna categoria, come Account AWS, servizi AWS o tipi di addebito specifici. Puoi, quindi, usare queste categorie in più prodotti nella AWS Billing and Cost Management e gestione dei costi AWS Billing and Cost Management. Sono inclusi AWS Cost Explorer, Budget AWS, AWS Cost and Usage Report e AWS Cost Anomaly Detection.

Come esempio, il diagramma seguente mostra in che modo raggruppare i costi e le informazioni sull'utilizzo nella tua organizzazione definendo più team (categoria di costo), molteplici ambienti (regole) e assegnando a ogni ambiente molteplici risorse o asset (dimensioni).



Organigramma relativo a costi e utilizzo

Puoi anche creare gruppi di costi con le categorie di costo. Dopo aver creato le categorie di costo (attendi fino a 24 ore dopo la creazione di una categoria di costo affinché i dati di utilizzo siano

aggiornati con i valori), appariranno in AWS Cost Explorer, Budget AWS, AWS Cost and Usage Reporte AWS Cost Anomaly Detection. In AWS Cost Explorer e Budget AWS, una categoria di costo appare come una componente di fatturazione aggiuntiva. Puoi usarla per filtrare valori specifici della categoria di costo o per definire i gruppi in base alla categoria di costo.

Passaggi dell'implementazione

- Definisci le categorie dell'organizzazione: organizza riunioni con le parti interessate interne e le unità di business per definire categorie che riflettano la struttura e i requisiti della tua organizzazione. Queste categorie dovrebbero corrispondere direttamente alla struttura delle categorie finanziarie esistenti, ad esempio unità aziendale, budget, centro di costi o reparto. Osserva i risultati che il cloud offre per la tua azienda, ad esempio la formazione o l'istruzione, poiché anche queste sono categorie organizzative.
- Definisci le categorie funzionali: organizza riunioni con le parti interessate interne e le unità di business per definire categorie che riflettano le funzioni presenti all'interno della tua azienda. Potrebbe trattarsi del carico di lavoro o dei nomi delle applicazioni e del tipo di ambiente, ad esempio produzione, test o sviluppo..
- Definisci le Categorie di costo AWS: Crea categorie di costi per organizzare le informazioni sui costi e sull'utilizzo mediante Categorie di costo AWS e mappa i costi e l'utilizzo di AWS in categorie significative. A una risorsa possono essere assegnate più categorie e una risorsa può essere in più categorie diverse, quindi definisci tutte le categorie necessarie in modo da essere in grado di gestire i costi all'interno della struttura categorizzata utilizzando le Categorie di costo AWS.

Risorse

Documenti correlati:

- Applicazione di tag alle risorse AWS
- Utilizzo dei tag di allocazione dei costi
- Analyzing your costs with Budget AWS
- Analisi dei costi con Cost Explorer
- Managing AWS Cost and Usage Reports
- Categorie di costo AWS
- Gestione dei costi con Categorie di costo AWS
- Creazione di categorie di costo
- Applicazione di tag alle categorie di costo

- Suddivisione degli addebiti all'interno delle categorie di costo
- Funzionalità delle Categorie di costo AWS

Esempi correlati:

- Organizza i dati su costi e utilizzo con Categorie di costo AWS
- Gestione dei costi con Categorie di costo AWS
- Well-Architected Labs: visualizzazione di costi e utilizzo
- Well-Architected Labs: Cost Categories

COST03-BP04 Definizione dei parametri dell'organizzazione

Definisci i parametri dell'organizzazione necessari per questo carico di lavoro. I parametri esemplificativi di un carico di lavoro sono i report dei clienti prodotti o le pagine web scaricate dai clienti.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

comprendi in che modo viene misurato l'output del carico di lavoro rispetto al successo aziendale. Ogni carico di lavoro ha in genere un piccolo set di output principali che indicano le prestazioni. Se disponi di un carico di lavoro complesso con molti componenti, puoi dare priorità alle voci dell'elenco o definire e monitorare i parametri per ogni componente. Collabora con i tuoi team per capire quali parametri utilizzare. Questa unità verrà utilizzata per comprendere l'efficienza del carico di lavoro o il costo per ciascun output aziendale.

Passaggi dell'implementazione

- Definisci i risultati del carico di lavoro: organizza riunioni con le parti interessate dell'azienda e definisci i risultati del carico di lavoro. Si tratta di una misura principale dell'utilizzo da parte dei clienti e devono essere parametri aziendali e non parametri tecnici. Deve esserci un piccolo numero di parametri di alto livello (meno di cinque) per carico di lavoro. Se il carico di lavoro produce più risultati per diversi casi d'uso, raggruppali in un singolo parametro.
- Definisci i risultati dei componenti del carico di lavoro: facoltativamente, se disponi di un carico di lavoro grande e complesso oppure puoi suddividere facilmente il carico di lavoro in componenti (ad esempio microservizi) con input e output ben definiti, definisci i parametri per ogni componente. Lo

sforzo deve riflettere il valore e il costo del componente. Inizia con i componenti più grandi e punta ai componenti più piccoli.

Risorse

Documenti correlati:

- Applicazione di tag alle risorse AWS
- Analisi dei costi con Budget AWS
- Analisi dei costi con Cost Explorer
- Gestione del report su costi e utilizzo di AWS

COST03-BP05 Configurazione degli strumenti di fatturazione e di gestione dei costi

Configura gli strumenti di gestione dei costi in conformità alle policy della tua organizzazione per gestire e ottimizzare gli investimenti nel cloud. Sono inclusi servizi, strumenti e risorse per organizzare e monitorare i dati su costi e utilizzo, migliorare il controllo tramite la fatturazione consolidata e le autorizzazioni di accesso, perfezionare la pianificazione tramite budget e previsioni, ricevere notifiche o avvisi e ridurre i costi tramite l'ottimizzazione di prezzi e risorse.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per definire consapevolezza e responsabilità forti (accountability), la strategia che interessa l'account deve essere considerata come parte integrante della strategia di allocazione dei costi. Definisci questo concetto ora per non doverlo affrontare in futuro. In caso contrario, il livello di consapevolezza potrebbe essere insufficiente e potrebbero verificarsi problemi in seguito.

Per incoraggiare la responsabilità degli investimenti nel cloud, fornisci agli utenti l'accesso a strumenti che offrono visibilità su costi e utilizzo. AWS consiglia di configurare tutti i carichi di lavoro e definire i team per i seguenti scopi:

- Organizzazione: definisci l'allocazione dei costi e i riferimenti della governance con la tua strategia di applicazione dei tag e la tua tassonomia. Crea più account AWS con strumenti come AWS Control Tower o AWS Organization. Applica i tag alle risorse AWS supportate e classificale in modo significativo in base alla struttura organizzativa (business unit, reparti o progetti). Applica i tag ai nomi degli account per centri di costo specifici e mappali con AWS Cost Categories per

raggruppare gli account delle business unit nei relativi centri di costo in modo che il responsabile della business unit possa visualizzare il consumo di più account in un'unica posizione.

- Accesso: monitora le informazioni di fatturazione a livello aziendale con la fatturazione consolidata e verifica che le parti interessate e i responsabili idonei abbiano accesso.
- Controllo: crea meccanismi di governance efficaci con i giusti guardrail per prevenire scenari imprevisti quando utilizzi le policy di controllo dei servizi (SCP), le policy di tag, le policy IAM e gli avvisi sul budget. Ad esempio, puoi consentire ai team di creare risorse specifiche nelle regioni preferite solo utilizzando meccanismi di controllo efficaci e impedire la creazione di risorse prive di tag specifici, come il centro di costo.
- Stato attuale: configura una dashboard che mostri i livelli correnti di costo e utilizzo. La dashboard deve essere disponibile in un luogo altamente visibile all'interno dell'ambiente di lavoro, in modo simile alla dashboard delle operazioni. Puoi esportare i dati e utilizzare la Dashboard costi e utilizzo dalla Centrale ottimizzazione costi AWS o qualsiasi prodotto supportato per creare questa visibilità. Potresti dover creare dashboard diverse per tipi di utenti diversi, ad esempio la dashboard per i manager sarà diversa dalla dashboard di progettazione.
- Notifiche: quando il costo o l'utilizzo superano i limiti definiti e si verificano anomalie, fornisci le notifiche con AWS Budgets o AWS Cost Anomaly Detection.
- Report: riepiloga tutte le informazioni su costi e utilizzo. Aumenta la consapevolezza e la responsabilità dei tuoi investimenti nel cloud con dati sui costi dettagliati e attribuibili. Crea i report con i suggerimenti pertinenti per il team che li utilizza.
- Monitoraggio: mostra i costi e l'utilizzo attuali rispetto a obiettivi o target stabiliti.
- Analisi: offri ai membri del team la possibilità di eseguire analisi personalizzate e approfondite fino alla granularità oraria, giornaliera o mensile con diversi filtri (risorsa, account, tag e così via).
- Esame: ricevi aggiornamenti sull'implementazione delle tue risorse e sull'opportunità di ottimizzazione dei costi. Ricevi le notifiche utilizzando Amazon CloudWatch, Amazon SNS o Amazon SES per le implementazioni delle risorse a livello di organizzazione. Esamina i suggerimenti per l'ottimizzazione dei costi con AWS Trusted Advisor o AWS Compute Optimizer.
- Report delle tendenze: mostra la variabilità dei costi e dell'utilizzo nel periodo richiesto e con la granularità necessaria.
- Previsioni: mostra i costi futuri stimati, prevedi l'utilizzo delle risorse e gestisci le spese con le dashboard delle previsioni create autonomamente.

Puoi usare la Centrale ottimizzazione costi AWS per comprendere le potenziali opportunità di risparmio sui costi consolidate da una posizione centralizzata e creare esportazioni di dati per

l'integrazione con Amazon Athena. Puoi utilizzare la Centrale ottimizzazione costi AWS anche per implementare la Dashboard costi e utilizzo, che usa Amazon QuickSight per l'analisi interattiva dei costi e la condivisione sicura degli approfondimenti sui costi.

Se non disponi delle competenze o della larghezza di banda essenziali nella tua organizzazione, puoi lavorare con AWS ProServ, AWS Managed Services (AMS) o partner AWS. Puoi anche utilizzare strumenti di terze parti, ma assicurati di convalidare la proposta di valore.

Passaggi dell'implementazione

- Consenti l'accesso agli strumenti basato sui team: configura i tuoi account e crea gruppi che abbiano accesso ai report richiesti su costi e utilizzo per i loro consumi e usa AWS Identity and Access Management per controllare l'accesso agli strumenti come AWS Cost Explorer. Questi gruppi devono includere i rappresentanti di tutti i team che possiedono o gestiscono un'applicazione. In questo modo si certifica che ogni team ha accesso alle informazioni sui costi e sull'utilizzo per tenere traccia dei propri consumi.
- Organizza tag e categorie di costo: organizza i costi tra team, business unit, applicazioni, ambienti e progetti. Usa i tag delle risorse per organizzare i costi, in base ai tag di allocazione dei costi. Crea le categorie di costo in base alle dimensioni utilizzando tag, account, servizi e così via per mappare i costi.
- Configura AWS Budgets: configura AWS Budgets per tutti gli account dei carichi di lavoro. Imposta un budget per la spesa complessiva dell'account e un budget per il carico di lavoro utilizzando i tag e le categorie di costo. Configura le notifiche in AWS Budgets per ricevere allarmi quando superi gli importi previsti nel budget o quando i costi stimati sono superiori a quelli dei tuoi budget.
- Configura il rilevamento delle anomalie dei costi AWS: usa il rilevamento delle anomalie dei costi AWS per gli account, i servizi principali o le categorie di costo che hai creato per monitorare costi e utilizzo e rilevare spese insolite. Puoi ricevere avvisi individualmente in report aggregati, oppure avvisi in un'email o in un argomento Amazon SNS per poter analizzare e stabilire il motivo principale di un’anomalia, nonché identificare il fattore che determina l'aumento dei costi.
- Usa gli strumenti di analisi dei costi: configura AWS Cost Explorer per il carico di lavoro e gli account e visualizza i dati sui costi per ulteriori analisi. Crea una dashboard per il carico di lavoro che tenga traccia della spesa generale e le metriche di utilizzo chiave per il carico di lavoro, nonché preveda i costi futuri sulla base dei tuoi dati storici.
- Utilizza gli strumenti di analisi per il risparmio sui costi: usa la Centrale ottimizzazione costi AWS per identificare le opportunità di risparmio con suggerimenti personalizzati, tra cui l'eliminazione delle risorse inutilizzate, il dimensionamento corretto, i Savings Plans, le prenotazioni e i suggerimenti del sistema di ottimizzazione del calcolo.

- Configura gli strumenti avanzati: puoi opzionalmente creare elementi visivi per facilitare l'analisi interattiva e la condivisione degli approfondimenti sui costi. Con le esportazioni dei dati della Centrale ottimizzazione costi AWS puoi creare per l'organizzazione una Dashboard costi e utilizzo basata su Amazon QuickSight con dettagli e granularità aggiuntivi. Puoi anche implementare funzionalità di analisi avanzate utilizzando le esportazioni dei dati in Amazon Athena per eseguire query avanzate e creare dashboard su Amazon QuickSight. Collabora con i partner AWS per adottare soluzioni di gestione del cloud per il monitoraggio e l'ottimizzazione della fatturazione consolidata del cloud.

Risorse

Documenti correlati:

- What is AWS Billing and Cost Management and Cost Management?
- Establishing your best practice AWS environment
- Best Practices for Tagging AWS Resources
- Tagging your AWS resources
- AWS Cost Categories
- Analyzing your costs with AWS Budgets
- Analyzing your costs with AWS Cost Explorer
- What is AWS Data Exports?

Video correlati:

- Deploying Cloud Intelligence Dashboards
- Get Alerts on any FinOps or Cost Optimization Metric or KPI

Esempi correlati:

- Cost and Usage Dashboard powered by Amazon QuickSight
- AWS Cost and Usage Governance Workshop

COST03-BP06 Allocazione dei costi in base alle metriche del carico di lavoro

Alloca i costi del carico di lavoro in base alle metriche di utilizzo o ai risultati aziendali per misurare l'efficienza dei costi del carico di lavoro. Implementa un processo per analizzare i dati relativi a costi e utilizzo con i servizi di analisi, che possono fornire informazioni approfondite e funzionalità di chargeback.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

Ottimizzare i costi significa conseguire i risultati aziendali al prezzo più basso eseguendo l'allocazione dei costi del carico di lavoro in base alle metriche di quest'ultimo, misurate in termini di efficienza. Monitora le metriche del carico di lavoro definite tramite file di log o altre funzionalità di monitoraggio dell'applicazione. Combina questi dati con i costi del carico di lavoro, che possono essere ottenuti osservando i costi con un determinato valore di tag o ID account. Esegui questa analisi a livello orario. L'efficienza cambia in genere se disponi di componenti di costo statico, come un database back-end sempre in esecuzione, con un tasso di richiesta variabile, ad esempio picchi di utilizzo tra le 9:00 e le 17:00 con poche richieste di notte. Comprendere la relazione tra i costi statici e i costi variabili ti aiuterà a rendere più mirate le tue attività di ottimizzazione.

La creazione di metriche del carico di lavoro per le risorse condivise può essere difficile rispetto a risorse come applicazioni containerizzate su Amazon Elastic Container Service (Amazon ECS) e Amazon API Gateway. Tuttavia, esistono alcuni modi per classificare l'utilizzo e tenere traccia dei costi. Se devi monitorare le risorse condivise di Amazon ECS e AWS Batch, puoi abilitare i dati di allocazione dei costi suddivisi in AWS Cost Explorer. Con i dati di allocazione dei costi suddivisi, puoi analizzare e ottimizzare i costi e l'utilizzo delle tue applicazioni containerizzate e riallocare i costi delle applicazioni alle singole entità aziendali in base al modo in cui vengono consumate le risorse di calcolo e memoria condivise.

Passaggi dell'implementazione

- Alloca i costi alle metriche del carico di lavoro: utilizzando le metriche definite e i tag configurati, crea una metrica che combini l'output e il costo del carico di lavoro. Utilizza i servizi di analisi come Amazon Athena e Amazon QuickSight per creare un pannello di controllo in grado di visualizzare l'efficienza del carico di lavoro complessivo e di ogni suo componente.

Risorse

Documenti correlati:

- Applicazione di tag alle risorse AWS
- Analisi dei costi con Budget AWS
- Analisi dei costi con Cost Explorer
- Gestione del report su costi e utilizzo di AWS

Esempi correlati:

- Improve cost visibility of Amazon ECS and AWS Batch with AWS Split Cost Allocation Data

COST 4. In che modo disattivi le risorse?

Implementa il controllo del cambiamento e la gestione delle risorse dall'inizio del progetto alla fine del ciclo di vita. In questo modo, puoi chiudere o interrompere le risorse non utilizzate per ridurre gli sprechi.

Best practice

- COST04-BP01 Monitoraggio delle risorse durante il loro ciclo di vita
- COST04-BP02 Implementazione di un processo di disattivazione
- COST04-BP03 Disattivazione delle risorse
- COST04-BP04 Disattivazione automatica delle risorse
- COST04-BP05 Applicare policy di conservazione dei dati

COST04-BP01 Monitoraggio delle risorse durante il loro ciclo di vita

Definisci e implementa un metodo per monitorare le risorse e le loro associazioni con i sistemi durante il loro ciclo di vita. Puoi usare l'applicazione di tag per identificare il carico di lavoro o la funzione della risorsa.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

Disattiva le risorse dei carichi di lavoro che non sono più necessarie. Un esempio comune sono le risorse utilizzate per i test: dopo il completamento dei test, le risorse possono essere rimosse. La tracciabilità delle risorse con i tag (e l'esecuzione di report su tali tag) può aiutare a identificare le risorse da disattivare, poiché non saranno più in uso o la loro licenza è in scadenza. L'utilizzo dei tag è un modo efficace per monitorare le risorse: puoi etichettare la risorsa con la relativa funzione o

con una data nota in cui può essere disattivata. Puoi quindi eseguire i report su questi tag. Esempi di valori per l'applicazione di tag relativi alle funzionalità sono `test funzionalità X` per identificare lo scopo della risorsa in termini di ciclo di vita del carico di lavoro. Un altro esempio è l'utilizzo di `LifeSpan` o `TTL` per le risorse, come il nome e il valore della chiave associati al tag da cancellare per definire il periodo di tempo al termine del quale deve avvenire la disattivazione o il momento specifico di tale attività.

Passaggi dell'implementazione

- Implementa uno schema di applicazione di tag: implementa uno schema di applicazione di tag che identifichi il carico di lavoro a cui appartiene la risorsa, verificando che tutte le risorse all'interno del carico di lavoro siano contrassegnate dai tag in modo conseguente. L'applicazione dei tag aiuta a classificare le risorse in base allo scopo, al team, all'ambiente o ad altri criteri rilevanti per l'azienda. Per maggiori dettagli sui casi d'uso, le strategie e le tecniche di applicazione dei tag, consulta Best practice per l'applicazione dei tag in AWS.
- Implementa il monitoraggio del throughput del carico di lavoro o degli output: Implementa il monitoraggio o gli allarmi del throughput del carico di lavoro, attivandolo sulle richieste in input o sulla restituzione dei risultati in output. Configuralo per fornire notifiche quando le richieste o gli output del carico di lavoro scendono a zero, indicando che le risorse del carico di lavoro non sono più utilizzate. Incorpora un fattore temporale se il carico di lavoro scende periodicamente a zero in condizioni normali. Per maggiori dettagli sulle risorse inutilizzate o sottoutilizzate, consulta Controllo per l'ottimizzazione dei costi di AWS Trusted Advisor.
- Raggruppa le risorse AWS: Crea gruppi per le risorse AWS. Puoi utilizzare AWS Resource Groups per ottimizzare e gestire le tue risorse AWS che si trovano nella stessa regione Regione AWS. Puoi aggiungere tag alla maggior parte delle risorse affinché sia possibile identificarle e ordinarle all'interno dell'organizzazione. Utilizza l'editor di tag per aggiungere tag alle risorse supportate in blocco. Valuta l'utilizzo di AWS Service Catalog per creare, gestire e distribuire agli utenti finali portafogli di prodotti approvati e gestire il ciclo di vita del prodotto.

Risorse

Documenti correlati:

- AWS Auto Scaling
- AWS Trusted Advisor
- Controlli per l'ottimizzazione dei costi di AWS Trusted Advisor
- Applicazione di tag alle risorse AWS

- Pubblicazione di parametri personalizzati

Video correlati:

- Come ottimizzare i costi utilizzando AWS Trusted Advisor

Esempi correlati:

- Organizza le risorse AWS
- Ottimizza i costi utilizzando AWS Trusted Advisor

COST04-BP02 Implementazione di un processo di disattivazione

Implementa un processo per identificare e disattivare le risorse inutilizzate.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

Implementa un processo standardizzato in tutta l'organizzazione per identificare e rimuovere le risorse inutilizzate. Il processo deve definire la frequenza di esecuzione della ricerca e i processi per rimuovere la risorsa al fine di verificare che tutti i requisiti dell'organizzazione siano soddisfatti.

Passaggi dell'implementazione

- Crea e implementa un processo di disattivazione: collabora con sviluppatori e proprietari del carico di lavoro alla creazione di un processo di disattivazione per il carico di lavoro e le relative risorse. Il processo deve includere il metodo per verificare se il carico di lavoro è in uso e quello per capire se ciascuna delle risorse del carico di lavoro è in uso. Specifica le fasi necessarie per disattivare la risorsa, rimuovendola dal servizio e garantendo allo stesso tempo la conformità a qualsiasi requisito normativo. Dovrebbero essere incluse tutte le risorse associate, come ad esempio le licenze o lo spazio di archiviazione collegato. Invia una notifica ai proprietari del carico di lavoro indicando che il processo di disattivazione è stato eseguito.

  Utilizza i seguenti passaggi di disattivazione per guidarti su quali dovrebbero essere le verifiche eseguite come parte del processo:

  - Identifica le risorse da disattivare: identifica le risorse che sono idonee alla disattivazione all'interno dell'ambiente Cloud AWS. Registra tutte le informazioni necessarie e pianifica la

disattivazione. Nella sequenza temporale, assicurati di tenere conto di eventuali problemi imprevisti e di quando si verificano durante il processo.

- Coordina e comunica: collabora con i proprietari del carico di lavoro per confermare le risorse da disattivare
- Registra i metadati e crea i backup: se necessario per le risorse nell'ambiente di produzione o se si tratta di risorse critiche, registra i metadati (come IP pubblici, regione, zona di disponibilità, VPC, sottorete e gruppi di sicurezza) e crea i backup (come snapshot Amazon Elastic Block Store o acquisizione di AMI, esportazione di chiavi ed esportazione di certificati).
- Valida la distribuzione come infrastructure-as-code: determina se le risorse sono state implementate utilizzando AWS CloudFormation, Terraform, AWS Cloud Development Kit (AWS CDK) o qualsiasi altro strumento di implementazione di infrastructure-as-code in modo che possano essere implementate di nuovo se necessario.
- Impedisci l'accesso: applica controlli restrittivi per un certo periodo di tempo al fine di impedire l'uso delle risorse mentre determini se la risorsa è necessaria. Verifica che l'ambiente delle risorse possa essere ripristinato allo stato originale, se necessario.
- Segui il processo di disattivazione interno: segui le attività amministrative e il processo di disattivazione dell'organizzazione, come la rimozione della risorsa dal dominio, la rimozione del record DNS e la rimozione della risorsa dagli strumenti di gestione della configurazione, di monitoraggio, di automazione e di sicurezza.

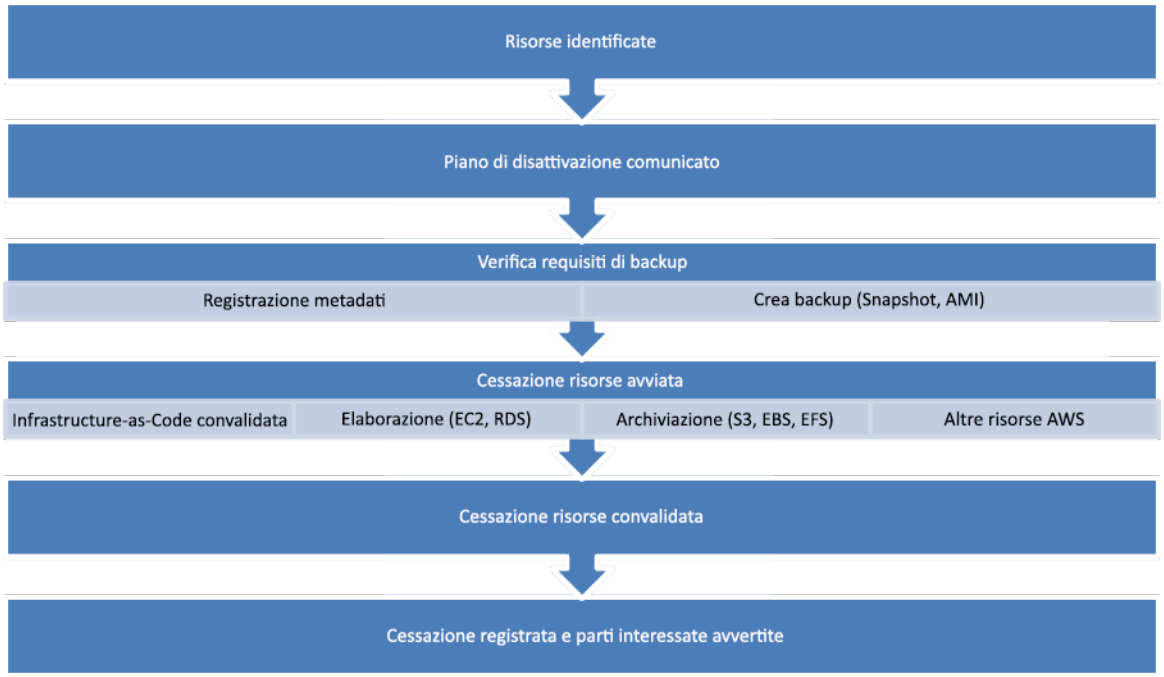
Se la risorsa è un'istanza Amazon EC2, consulta l'elenco seguente. Per ulteriori dettagli, consulta In che modo è possibile eliminare o terminare le risorse di Amazon EC2?

- Arresta o termina tutte le tue istanze Amazon EC2 e i sistemi di bilanciamento del carico. Per un breve periodo dopo la loro eliminazione le istanze Amazon EC2 continuano a essere visibili nella console. Non verrà addebitato alcun costo per le istanze che non si trovano in stato di esecuzione
- Elimina l'infrastruttura Auto Scaling.
- Rilascia tutti gli host dedicati.
- Elimina tutti i volumi e gli snapshot Amazon EBS.
- Rilascia tutti gli Indirizzi IP elastici.
- Annulla la registrazione di tutte le Amazon Machine Image (AMI).
- Terminata tutti gli ambienti AWS Elastic Beanstalk.

Se la risorsa è un oggetto in uno spazio di archiviazione Amazon S3 Glacier e se si elimina un archivio prima di aver raggiunto la durata minima di archiviazione, verrà addebitato un costo

di eliminazione anticipata proporzionale. La durata minima di archiviazione di Amazon S3 Glacier dipende dalla classe di archiviazione utilizzata. Per un riepilogo della durata minima di archiviazione per ciascuna classe di archiviazione, consulta Prestazioni delle classi di archiviazione Amazon S3. Per informazioni dettagliate sulle modalità di calcolo delle tariffe di cancellazione anticipata, consulta Prezzi di Amazon S3.

Il seguente semplice diagramma di flusso del processo di disattivazione illustra le fasi della disattivazione. Prima di disattivare le risorse, verifica che le risorse identificate per la disattivazione non siano utilizzate dall'organizzazione.



Flusso del processo di disattivazione delle risorse

Risorse

Documenti correlati:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS CloudTrail

Video correlati:

- Cancella lo stack CloudFormation ma mantieni alcune risorse

- Scopri quale utente ha avviato l'istanza Amazon EC2

Esempi correlati:

- Elimina o termina le risorse Amazon EC2
- Scopri quale utente ha avviato l'istanza Amazon EC2

COST04-BP03 Disattivazione delle risorse

Disattivazione delle risorse attivate da eventi come audit periodici o modifiche relative all'utilizzo. La disattivazione viene in genere eseguita periodicamente e può essere manuale o automatizzata.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La frequenza e lo sforzo di ricerca delle risorse inutilizzate dovrebbero riflettere i risparmi potenziali, pertanto un account con costi contenuti deve essere analizzato con una frequenza minore rispetto a un account che ha costi maggiori. Gli eventi di ricerca e disattivazione possono essere attivati da modifiche di stato nel carico di lavoro, ad esempio il termine del ciclo di vita di un prodotto o la sua sostituzione. Le ricerche e gli eventi di disattivazione possono anche essere attivati da eventi esterni, ad esempio cambiamenti nelle condizioni di mercato o cessazione del prodotto.

Passaggi dell'implementazione

- Disattivazione delle risorse: si tratta della fase di disattivazione delle risorse AWS non più necessarie o del termine di un contratto di licenza. Completa tutti i controlli finali prima di passare alla fase di dismissione e disattivazione delle risorse per evitare interruzioni indesiderate durante fasi come l'esecuzione di snapshot o backup. Utilizzando il processo di disattivazione, disattiva tutte le risorse identificate come inutilizzate.

Risorse

Documenti correlati:

- AWS Auto Scaling
- AWS Trusted Advisor

Esempi correlati:

- Well-Architected Labs: disattivazione delle risorse (Livello 100)

COST04-BP04 Disattivazione automatica delle risorse

Progetta il tuo carico di lavoro in modo da gestire in modo controllato la disattivazione delle risorse, identificando e disattivando le risorse non critiche, le risorse non necessarie o quelle a basso utilizzo.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Utilizza l'automazione per ridurre o rimuovere i costi associati al processo di ritiro. Progettare il carico di lavoro per eseguire automaticamente la disattivazione ridurrà i costi complessivi del carico di lavoro durante il suo ciclo di vita. Per eseguire il processo di disattivazione puoi utilizzare AWS Auto Scaling. Puoi anche implementare un codice personalizzato utilizzando l'API o l'SDK per disattivare automaticamente le risorse associate a un carico di lavoro.

Le applicazioni moderne sono sviluppate in modalità serverless-first, una strategia che dà priorità all'adozione di servizi serverless. AWS ha sviluppato servizi serverless per tutti e tre i livelli dello stack: calcolo, integrazione e memorizzazione dei dati. L'utilizzo di un'architettura serverless consente di risparmiare sui costi nei periodi di scarso traffico e di approfittare del dimensionamento automatico.

Passaggi dell'implementazione

- Implementa AWS Auto Scaling: nel caso delle risorse che sono supportate, configurale con AWS Auto Scaling. AWS Auto Scaling può aiutarti a ottimizzare l'utilizzo e l'efficienza dei costi durante l'utilizzo dei servizi AWS. Quando la domanda diminuisce, AWS Auto Scaling rimuove automaticamente la capacità di risorse in eccesso per evitare spese inutili.
- Configura CloudWatch per la terminazione delle istanze: le istanze possono essere configurate affinché terminino in base agli allarmi CloudWatch. Utilizzando i parametri del processo di disattivazione, implementa un allarme con un'operazione Amazon Elastic Compute Cloud. Verifica l'operazione in un ambiente non di produzione prima di eseguire il roll out.
- Implementa il codice all'interno del carico di lavoro per disattivare le risorse associate al carico di lavoro puoi utilizzare l'SDK AWS o la AWS CLI. Implementa il codice all'interno dell'applicazione che si integra con AWS e termina o rimuove le risorse che non vengono più utilizzate.
- Utilizza servizi serverless: per compilare ed eseguire le tue applicazioni dai la priorità allo sviluppo di architetture serverless e architetture basate su eventi su AWS. AWS offre diversi servizi basati su tecnologie serverless che offrono un utilizzo intrinsecamente ottimizzato delle risorse e la

disattivazione automatizzata (riduzione e incremento orizzontali). Con le applicazioni serverless, l'utilizzo delle risorse viene ottimizzato automaticamente e non si paga mai il provisioning in eccesso.

Risorse

Documenti correlati:

- AWS Auto Scaling
- AWS Trusted Advisor
- Serverless su AWS
- Crea allarmi per arrestare, terminare, riavviare o recuperare un'istanza
- Nozioni di base su Amazon EC2 Auto Scaling
- Aggiungi le operazioni di terminazione agli allarmi Amazon CloudWatch

Esempi correlati:

- Pianificazione della cancellazione automatica degli stack AWS CloudFormation
- Well-Architected Labs: disattivazione delle risorse (Livello 100)
- Pulizia automatica su AWS di Servian

COST04-BP05 Applicare policy di conservazione dei dati

Definisci le policy di conservazione dei dati su risorse supportate per gestire l'eliminazione degli oggetti in base ai requisiti della tua organizzazione. Identifica ed elimina risorse non necessarie oppure orfane e oggetti non più richiesti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Usa le policy di conservazione dei dati e del ciclo di vita per ridurre i costi associati al processo di disattivazione e i costi di archiviazione per le risorse identificate. La definizione delle policy di conservazione dei dati e del ciclo di vita per eseguire l'eliminazione e la migrazione di classi di archiviazione automatizzate contribuirà a ridurre i costi di archiviazione generale durante la sua durata. Si può usare Amazon Data Lifecycle Manager per automatizzare la creazione e l'eliminazione di snapshot Amazon Elastic Block Store e Amazon Machine Images (AMI) supportate da Amazon EBS e usare Amazon S3 Intelligent-Tiering o una configurazione del ciclo di vita Amazon S3

per gestire il ciclo di vita dei tuoi oggetti Amazon S3. È possibile anche implementare un codice personalizzato utilizzando un'API o un SDK per creare policy del ciclo di vita e regole di policy per oggetti da eliminare automaticamente.

Passaggi dell'implementazione

- Usa Amazon Data Lifecycle Manager: usa policy del ciclo di vita su Amazon Data Lifecycle Manager per automatizzare l'eliminazione di snapshot Amazon EBS e AMI supportate da Amazon EBS.
- Imposta la configurazione del ciclo di vita su un bucket: usa la configurazione del ciclo di vita di Amazon S3 su un bucket per definire le azioni che Amazon S3 deve intraprendere durante il ciclo di vita di un oggetto, oltre all'eliminazione alla fine del ciclo di vita di un oggetto, in base ai requisiti aziendali.

Risorse

Documenti correlati:

- AWS Trusted Advisor
- Amazon Data Lifecycle Manager
- Come impostare la configurazione del ciclo di vita su un bucket Amazon S3

Video correlati:

- Automatizzare gli snapshot Amazon EBS con Amazon Data Lifecycle Manager
- Svuotare un bucket Amazon S3 con una regola di configurazione del ciclo di vita

Esempi correlati:

- Svuotare un bucket Amazon S3 con una regola di configurazione del ciclo di vita
- Well-Architected Labs: disattivazione automatica delle risorse (Livello 100)

Risorse a costi contenuti

Domande

- COST 5. In che modo valuti i costi quando selezioni i servizi?

- COST 6. In che modo raggiungi gli obiettivi di costo quando selezioni il tipo, le dimensioni e il numero delle risorse?
- COST 7. In che modo impieghi i modelli di prezzo per ridurre i costi?
- COST 8. In che modo pianifichi i costi per il trasferimento dei dati?

COST 5. In che modo valuti i costi quando selezioni i servizi?

Amazon EC2, Amazon EBS e Amazon S3 sono servizi AWS di base. I servizi gestiti, come Amazon RDS e Amazon DynamoDB, sono servizi AWS di livello superiore o applicativo. Selezionando i blocchi predefiniti e i servizi gestiti appropriati, è possibile ottimizzare questo carico di lavoro per i costi. Ad esempio, utilizzando i servizi gestiti, puoi ridurre o eliminare gran parte dei costi generali amministrativi e operativi, liberandotene per lavorare su applicazioni e attività correlate al tuo business.

Best practice

- COST05-BP01 Identificazione dei requisiti dell'organizzazione sui costi
- COST05-BP02 Analisi di tutti i componenti del carico di lavoro
- COST05-BP03 Esecuzione di un'analisi accurata di ciascun componente
- COST05-BP04 Selezione di software con licenze convenienti
- COST05-BP05 Selezione dei componenti del carico di lavoro per ottimizzare i costi in linea con le priorità dell'organizzazione
- COST05-BP06 Esecuzione di un'analisi dei costi per diversi valori di utilizzo nel tempo

COST05-BP01 Identificazione dei requisiti dell'organizzazione sui costi

Lavora con i membri del team per definire il bilanciamento tra l'ottimizzazione dei costi e altri pilastri, come le prestazioni e l'affidabilità, per questo carico di lavoro.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

Nella maggior parte delle organizzazioni, il reparto di tecnologia dell'informazione (IT) è composto da diversi team di piccole dimensioni, ciascuno con una propria agenda e area di interesse, che riflettono le specializzazioni e le competenze dei suoi membri. È necessario comprendere gli obiettivi, le priorità e le finalità generali dell'organizzazione e in che modo ogni reparto o progetto contribuisce a

tali obiettivi. La catalogazione di tutte le risorse essenziali, inclusi personale, attrezzature, tecnologia, materiali e servizi esterni, è fondamentale per il raggiungimento degli obiettivi organizzativi e una pianificazione precisa del budget. L'adozione di questo approccio sistematico all'identificazione e alla comprensione dei costi è fondamentale per stabilire un piano dei costi realistico e affidabile per l'organizzazione.

al momento di selezionare i servizi per un carico di lavoro, è fondamentale comprendere le priorità dell'organizzazione. Assicurati che vi sia equilibrio tra i costi e gli altri pilastri del Framework AWS Well-Architected, ad esempio prestazioni e affidabilità. È necessario eseguire questo processo sistematicamente e regolarmente in modo da acquisire i cambiamenti a livello di obiettivi, condizioni di mercato e dinamiche operative dell'organizzazione. Un carico di lavoro completamente ottimizzato per i costi è la soluzione più in linea con i requisiti della tua organizzazione, e non necessariamente quella con il costo più basso. Per raccogliere il maggior numero di informazioni, interpella tutti i team all'interno dell'organizzazione, come i team dedicati ai prodotti, di business, tecnici e finanziari. Valuta l'impatto dei compromessi tra interessi concorrenti o approcci alternativi, per aiutare a prendere decisioni informate quando si stabilisce dove concentrare le attività o scegliere una linea di azione.

Ad esempio, accelerare l'introduzione sul mercato di nuove funzionalità può essere preferibile all'ottimizzazione dei costi. Oppure, è possibile scegliere un database relazionale per i dati non relazionali per semplificare la migrazione di un sistema, anziché migrare a un database ottimizzato per il tuo tipo di dati e aggiornare l'applicazione.

Passaggi dell'implementazione

- Identifica i requisiti dell'organizzazione sui costi: organizza riunioni con i membri dei team della tua organizzazione, tra cui i team di gestione dei prodotti, i team proprietari delle applicazioni, i team operativi e di sviluppo, i team di gestione e finanziari. Dai la priorità ai pilastri Well-Architected per questo carico di lavoro e i relativi componenti. L'output dovrebbe essere un elenco ordinato dei pilastri. Puoi anche aggiungere un fattore di ponderazione a ciascun pilastro per indicare il livello di attenzione aggiuntiva assegnato o quanto è simile il livello di attenzione assegnato a due pilastri.
- Analizza il debito tecnico e documentalo: durante la revisione del carico di lavoro, analizza il debito tecnico. Documenta gli elementi lasciati in sospeso per riesaminare il carico di lavoro in un secondo momento, con l'obiettivo di rifattorizzarlo o riprogettarlo per ottimizzarlo ulteriormente. Alle altre parti interessate è fondamentale comunicare in modo chiaro le scelte di compromesso adottate.

Risorse

Best practice correlate:

- REL11-BP07 Progettazione del prodotto in modo da soddisfare gli obiettivi di disponibilità e i contratti sul livello di servizio per i tempi di attività
- OPS01-BP06 Valutazione dei compromessi

Documenti correlati:

- Calcolatore del costo totale di proprietà (TCO) di AWS
- Classi di archiviazione di Amazon S3
- Prodotti cloud

COST05-BP02 Analisi di tutti i componenti del carico di lavoro

Verifica che ogni componente del carico di lavoro venga analizzato, indipendentemente dalle dimensioni attuali o dai costi correnti. L'attività di revisione deve riflettere i potenziali benefici, come i costi correnti e quelli previsti.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

I componenti del carico di lavoro, progettati per fornire valore aziendale all'organizzazione, possono includere vari servizi. Per ogni componente, è possibile scegliere servizi Cloud AWS specifici per soddisfare specifiche esigenze aziendali. Questa selezione potrebbe essere influenzata da fattori quali la familiarità o l'esperienza precedente nell'uso di questi servizi.

Dopo aver identificato i requisiti dell'organizzazione, come indicato in COST05-BP01 Identificazione dei requisiti dell'organizzazione sui costi, esegui un'analisi approfondita di tutti i componenti del carico di lavoro. Analizza ogni componente considerando i costi e le dimensioni attuali e previsti. Considera il costo dell'analisi rispetto a qualsiasi potenziale risparmio del carico di lavoro durante il suo ciclo di vita. L'impegno dedicato all'analisi di tutti i componenti di questo carico di lavoro deve corrispondere al potenziale risparmio o ai miglioramenti previsti derivanti dall'ottimizzazione del componente specifico. Ad esempio, se il costo della risorsa proposta è di 10 USD al mese e secondo le previsioni i carichi non dovrebbero superare i 15 USD al mese, spendere un giorno di lavoro per ridurre i costi del 50% (5 USD al mese) potrebbe eccedere il potenziale beneficio nel corso della vita del sistema. Usa una stima basata sui dati, più rapida ed efficiente, per generare il migliore risultato complessivo per questo componente.

I carichi di lavoro possono cambiare nel corso del tempo e il giusto set di servizi potrebbe non essere ottimale se l'architettura o l'utilizzo del carico di lavoro cambiano. L'analisi per la selezione dei servizi deve integrare gli stati del carico di lavoro e i livelli di utilizzo attuali e futuri. Implementare un servizio in funzione dello stato o dell'utilizzo futuro del carico di lavoro può ridurre i costi complessivi, diminuendo o rimuovendo l'impegno necessario per apportare modifiche future. Ad esempio, EMR Serverless potrebbe inizialmente essere la scelta appropriata. Tuttavia, con l'aumento del consumo del servizio, il passaggio a EMR su EC2 potrebbe ridurre i costi per il componente specifico del carico di lavoro.

AWS Cost Explorer e AWS Cost and Usage Report (CUR) possono analizzare i costi di una proof of concept (PoC) o di un ambiente in esecuzione. Puoi anche utilizzare AWS Pricing Calculator per stimare i costi del carico di lavoro.

Scrivi un flusso di lavoro che dovrà essere usato dai team tecnici per esaminare i carichi di lavoro. Il flusso di lavoro deve essere semplice, ma coprire tutti i passaggi necessari affinché i team comprendano ogni componente del carico di lavoro e i relativi prezzi. L'organizzazione può quindi seguire e personalizzare il flusso di lavoro in base alle esigenze specifiche di ogni team.

1. Elenca ogni servizio in uso per il tuo carico di lavoro: questo è un buon punto di partenza. Identifica tutti i servizi attualmente in uso e da dove derivano i costi.
2. Comprendi come funzionano i prezzi per questi servizi: esamina il modello di prezzi di ciascun servizio. Servizi AWS diversi hanno modelli di prezzi diversi in base a fattori come il volume di utilizzo, il trasferimento dei dati e i prezzi specifici delle funzionalità.
3. Concentrati sui servizi che comportano costi di carico di lavoro imprevisti e che non sono in linea con l'utilizzo previsto e il risultato aziendale: individua i valori anomali o i servizi in cui il costo non è proporzionale al valore o all'utilizzo con AWS Cost Explorer o AWS Cost and Usage Report. È importante correlare i costi ai risultati aziendali per poter definire le priorità delle attività di ottimizzazione.
4. Usa AWS Cost Explorer, CloudWatch Logs, Log di flusso VPC e Amazon S3 Storage Lens per comprendere la causa principale dei costi elevati: questi strumenti sono fondamentali nella diagnosi dei costi elevati. Ogni servizio offre una visione diversa per osservare e analizzare l'utilizzo e i costi. Ad esempio, Cost Explorer aiuta a determinare le tendenze generali dei costi, CloudWatch Logs fornisce approfondimenti operativi, VPC Flow Logs mostra il traffico IP e Amazon S3 Storage Lens è utile per l'analisi dell'archiviazione.
5. Usa Budget AWS per impostare i budget per determinati servizi o account: l'impostazione dei budget è un modo proattivo per gestire i costi. Utilizza Budget AWS per definire soglie di budget personalizzate e ricevere avvisi quando i costi superano tali soglie.

6. Configura gli allarmi Amazon CloudWatch per inviare avvisi di fatturazione e utilizzo: configura il monitoraggio e gli avvisi per le metriche di costi e utilizzo. Gli allarmi CloudWatch possono avvisarti quando vengono superate determinate soglie, migliorando i tempi di risposta dell'intervento.

Favorisci notevoli miglioramenti e risparmi finanziari nel tempo con la revisione strategica di tutti i componenti del carico di lavoro, indipendentemente dalle caratteristiche attuali. L'impegno profuso in questo processo di revisione deve essere ponderato, con un'attenta considerazione dei potenziali vantaggi che si possono ottenere.

Passaggi dell'implementazione

- Elenca i componenti del carico di lavoro: crea un elenco dei componenti del carico di lavoro. Usa questo elenco per verificare che ogni componente sia stato analizzato. Gli impegni sostenuti devono riflettere la criticità del carico di lavoro secondo quanto definito dalle priorità dell'organizzazione. Raggruppa le risorse in modo funzionale, ad esempio in base all'archiviazione del database di produzione, per migliorare l'efficienza se sono presenti più database.
- Assegna le priorità all'elenco dei componenti: assegna ai componenti in elenco una priorità in base all'impegno che richiedono. Tale assegnazione viene in genere eseguita in ordine dal componente più costoso a quello meno costoso o in base alla criticità definita dalle priorità dell'organizzazione.
- Esegui l'analisi: per ogni componente dell'elenco, esamina le opzioni e i servizi disponibili e scegli l'opzione che si allinea meglio alle priorità dell'organizzazione.

Risorse

Documenti correlati:

- AWS Pricing Calculator
- AWS Cost Explorer
- Classi di archiviazione Amazon S3
- Prodotti Cloud AWS

Video correlati:

- AWS Cost Optimization Series: CloudWatch

COST05-BP03 Esecuzione di un'analisi accurata di ciascun componente

Considera il costo complessivo per l'organizzazione di ogni componente. Considera il costo totale di proprietà tenendo conto dei costi operativi e di gestione, soprattutto quando si utilizzano i servizi gestiti del provider cloud. L'attività di revisione deve riflettere i potenziali benefici (ad esempio il tempo speso per l'analisi dovrebbe essere proporzionale al costo dei componenti).

Livello di rischio associato se questa best practice non fosse adottata: Alta

Guida all'implementazione

Si consideri tempo risparmiato, che consentirà al proprio team di concentrarsi sull'eliminazione del debito tecnico, sull'innovazione e sulle funzionalità che offrono un valore aggiunto e sullo sviluppo di ciò che diversifica il business. Ad esempio, si potrebbe avere la necessità di eseguire il rehosting (lift and shift) del proprio database dall'ambiente on-premise nel cloud il più rapidamente possibile ed eseguire l'ottimizzazione in un secondo momento. Vale la pena soffermarsi sul risparmio possibile che puoi ottenere usando i servizi gestiti su AWS che rimuovono o riducono i costi di licenza. I servizi gestiti su AWS eliminano l'onere operativo e amministrativo legato alla manutenzione di un servizio, come l'applicazione di patch o l'aggiornamento del sistema operativo, consentendoti di concentrarti sull'innovazione e sul business.

Dato che i servizi gestiti operano su scala cloud, possono offrire un costo inferiore per transazione o servizio. Questo vuol dire fare alcune ottimizzazioni potenziali in modo da ottenere benefici tangibili, senza modificare l'architettura principale dell'applicazione. Ad esempio, si potrebbe voler ridurre il tempo dedicato alla gestione delle istanze di database migrando verso una piattaforma di database-as-a-service come Amazon Relational Database Service (Amazon RDS) o migrando la propria applicazione a una piattaforma completamente gestita come AWS Elastic Beanstalk.

Solitamente, i servizi gestiti presentano attributi che si possono impostare per garantire la capacità necessaria. Devi impostare e monitorare questi attributi in modo che la tua capacità in eccesso sia mantenuta al minimo e le prestazioni siano massimizzate. Puoi modificare gli attributi di AWS Managed Services utilizzando la AWS Management Console o le API e gli SDK AWS per allineare le risorse necessarie con le variazioni della domanda. Ad esempio, puoi aumentare o diminuire il numero di nodi di un cluster Amazon EMR (o di un cluster Amazon Redshift) per ridimensionarlo.

Puoi anche unire più istanze in una risorsa AWS per ottenere una densità di utilizzo più elevata. Ad esempio, puoi effettuare il provisioning di diversi database più piccoli su una singola istanza database Amazon Relational Database Service (Amazon RDS). Quando l'utilizzo si intensifica, puoi migrare uno dei database su un'istanza database Amazon RDS dedicata utilizzando un processo di generazione dello snapshot e ripristino.

Quando predisponi carichi di lavoro su servizi gestiti, devi comprendere i requisiti inerenti alla modifica della capacità del servizio. Tali requisiti solitamente riguardano il tempo, l'impegno e qualunque impatto sul normale funzionamento del carico di lavoro. La risorsa predisposta deve offrire il tempo necessario per l'applicazione delle modifiche, pertanto procurati i mezzi necessari a tal fine. L'impegno costante richiesto per modificare i servizi può essere ridotto praticamente a zero grazie alle API e agli SDK integrati con strumenti di sistema e di monitoraggio come Amazon CloudWatch.

Amazon RDS, Amazon Redshift, e Amazon ElastiCache offrono un servizio di database gestito. Amazon Athena, Amazon EMR e Amazon OpenSearch Service offrono un servizio di analisi gestito.

AMS è un servizio che gestisce l'infrastruttura AWS per conto di clienti e partner aziendali. Offre un ambiente sicuro e conforme su cui è possibile implementare i carichi di lavoro. AMS utilizza modelli operativi cloud aziendali dotati di automazione per consentirti di soddisfare i requisiti aziendali, di passare più rapidamente al cloud e di ridurre i costi di gestione correnti.

Passaggi dell'implementazione

- Esegui un'analisi completa: utilizzando l'elenco dei componenti, analizza ogni componente dalla priorità più alta alla priorità più bassa. Per la priorità più alta e i componenti più costosi, esegui analisi aggiuntive e valuta tutte le opzioni disponibili e il loro impatto a lungo termine. Per i componenti con priorità più bassa, valuta se le modifiche relative all'utilizzo hanno un impatto sulla priorità del componente, quindi esegui un'analisi dello sforzo appropriato.
- Confronta risorse gestite e non gestite: considera i costi operativi delle risorse che gestisci e confrontali con quelli delle risorse gestite AWS. Ad esempio, rivedi i tuoi database in esecuzione su istanze Amazon EC2 e confrontali con le opzioni Amazon RDS (un servizio gestito AWS) o Amazon EMR paragonato all'esecuzione di Apache Spark su Amazon EC2. Quando si passa da un carico di lavoro autogestito a un carico di lavoro AWS completamente gestito, esamina attentamente le tue opzioni. I tre fattori più importanti da considerare sono il tipo di servizio gestito che vuoi usare, il processo che userai per migrare i tuoi dati e comprendere il AWS modello di responsabilità condivisa.

Risorse

Documenti correlati:

- Calcolatore del costo totale di proprietà (TCO) di AWS
- Classi di archiviazione di Amazon S3
- Prodotti Cloud AWS

- Modello di responsabilità condivisa AWS

Video correlati:

- Perché passare a un database gestito?
- Che cos'è Amazon EMR e come posso usarlo per elaborare i dati?

Esempi correlati:

- Perché passare a un database gestito
- Consolida i dati da database SQL Server identici in un unico database Amazon RDS for SQL Server con AWS DMS
- Distribuisci i dati su scala su Amazon Managed Streaming for Apache Kafka (Amazon MSK)
- Migra un'applicazione Web ASP.NET su AWS Elastic Beanstalk

COST05-BP04 Selezione di software con licenze convenienti

Il software open source elimina i costi di licenza del software, che contribuiscono in modo significativo ai costi dei carichi di lavoro. Nei casi in cui il software con licenza sia obbligatorio, evita le licenze legate ad attributi arbitrari, ad esempio CPU, e cerca le licenze legate all'output o ai risultati. Il costo di queste licenze si ridimensiona in base ai vantaggi che offrono.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

Il concetto di open source è nato nel contesto dello sviluppo del software per indicare che il software è conforme a determinati criteri di distribuzione gratuita. Il software open source è composto da codice sorgente che chiunque può analizzare, modificare e migliorare. In base ai requisiti aziendali, alle competenze professionali, all'utilizzo previsto o ad altre dipendenze tecnologiche, le organizzazioni possono prendere in considerazione l'utilizzo di software open source in AWS per ridurre al minimo i costi di licenza. In altre parole, utilizzando software open source è possibile eliminare il costo delle licenze software. Con l'aumentare delle dimensioni del carico di lavoro, l'impatto sui costi può essere significativo.

Misura i vantaggi di usare software con licenza in rapporto ai costi totali per ottimizzare il carico di lavoro. Crea modelli per le eventuali modifiche alla licenza e il relativo impatto sui costi del carico

di lavoro. Se un fornitore modifica il costo della licenza del database, valuta come questo incide sull'efficienza complessiva del carico di lavoro. Effettua un'analisi dello storico dei prezzi dei tuoi fornitori per scoprire le tendenze dei cambiamenti relativi alle licenze dei loro prodotti. I costi delle licenze possono anche essere adattati indipendentemente dal throughput o dall'utilizzo, come nel caso delle licenze che si adattano in base all'hardware (licenze legate alla CPU). È necessario evitare queste licenze poiché i costi possono aumentare rapidamente senza che vi siano vantaggi correlati.

Ad esempio, l'utilizzo di un'istanza Amazon EC2 in us-east-1 con un sistema operativo Linux consente di ridurre i costi di circa il 45% rispetto all'esecuzione di un'altra istanza Amazon EC2 eseguita su Windows.

AWS Pricing Calculator offre un modo completo per confrontare i costi di varie risorse con diverse opzioni di licenza, come le istanze Amazon RDS e diversi motori di database. Inoltre, AWS Cost Explorer fornisce un punto di vista impareggiabile per i costi dei carichi di lavoro esistenti, in particolare quelli derivanti da licenze diverse. Per la gestione delle licenze, AWS License Manager offre un metodo semplificato per supervisionare e gestire le licenze software. I clienti possono implementare e rendere operativo il loro software open source preferito nel Cloud AWS.

Passaggi dell'implementazione

- Analizza le opzioni di licenza: rivedi i termini di licenza del software disponibile. Cerca le versioni open source che dispongono delle funzionalità necessarie e considera se i vantaggi del software con licenza superano i costi. Condizioni convenienti possono rendere il costo del software proporzionato ai vantaggi che offre.
- Analizza il fornitore del software: esamina tutte le modifiche ai prezzi o alle licenze apportate dal fornitore. Identifica eventuali modifiche non allineate ai risultati, ad esempio termini punitivi per l'esecuzione su hardware o piattaforme di fornitori specifici. Inoltre, verifica il modo in cui vengono eseguiti gli audit e le sanzioni in cui potresti incorrere.

Risorse

Documenti correlati:

- Open Source at AWS
- Calcolatore del costo totale di proprietà (TCO) di AWS
- Classi di archiviazione di Amazon S3
- Prodotti cloud

Esempi correlati:

- Blog relativi all'open source
- Blog relativi all'open source AWS
- Optimization and Licensing Assessment

COST05-BP05 Selezione dei componenti del carico di lavoro per ottimizzare i costi in linea con le priorità dell'organizzazione

Tieni in considerazione il costo nella selezione di tutti i componenti del tuo carico di lavoro. Ciò include l'utilizzo di servizi a livello di applicazione e servizi gestiti o serverless, container o un'architettura basata sugli eventi per ridurre i costi complessivi. Riduci al minimo i costi di licenza utilizzando software open source, software che non hanno costi di licenza o altre alternative per contenere la spesa.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Quando si selezionano tutti i componenti, è necessario considerare il costo dei servizi e delle opzioni. Questo include l'utilizzo di servizi gestiti e a livello di applicazione, come Amazon Relational Database Service (Amazon RDS), Amazon DynamoDB, Amazon Simple Notification Service (Amazon SNS) e Amazon Simple Email Service (Amazon SES) per ridurre il costo complessivo dell'organizzazione.

Utilizza funzioni serverless e container per l'elaborazione, come AWS Lambda e Amazon Simple Storage Service (Amazon S3) per i siti web statici. Se possibile, containerizza la tua applicazione e utilizza servizi di container gestiti di AWS come Amazon Elastic Container Service (Amazon ECS) oppure Amazon Elastic Kubernetes Service (Amazon EKS).

Riduci al minimo i costi di licenza utilizzando software open source o software che non prevedono tariffe di licenza, come ad esempio Amazon Linux per carichi di lavoro di calcolo, oppure esegui la migrazione dei database su Amazon Aurora.

puoi utilizzare servizi serverless o a livello di applicazione, ad esempio Lambda, Amazon Simple Queue Service (Amazon SQS), Amazon SNSe Amazon SES. Questi servizi eliminano la necessità di gestire una risorsa e forniscono funzioni di esecuzione del codice, servizi di accodamento e consegna dei messaggi. L'altro vantaggio è che le prestazioni e i costi vengono adattati in base all'utilizzo, garantendo l'allocazione e l'attribuzione dei costi in modo efficiente.

Utilizzando un'architettura basata su eventi è possibile anche con servizi serverless. Le architetture basate su eventi funzionano su base push, per cui tutto succede on demand quando l'evento si presenta sul router. In questo modo non devi sostenere i costi di un continuo polling per verificare un evento. Ciò significa minor consumo di larghezza di banda della rete, minor utilizzo della CPU, minor capacità di parco istanze inattiva e minor numero di handshake SSL/TLS.

Per ulteriori informazioni sul serverless, consultare whitepaper di approfondimento sulle applicazioni serverless secondo il Canone di architettura.

Passaggi dell'implementazione

- Seleziona ciascun servizio per ottimizzare i costi: Utilizzando l'elenco e l'analisi prioritari, seleziona ogni opzione che fornisce la migliore corrispondenza con le priorità dell'organizzazione. Invece di aumentare la capacità per soddisfare la domanda, prendi in considerazione altre opzioni che potrebbero offrirti performance migliori a costi inferiori. Ad esempio, è necessario rivedere il traffico previsto per i database su AWS e prendere in considerazione la possibilità di aumentare le dimensioni dell'istanza o di utilizzare servizi Amazon ElastiCache (Redis o Memcached) per fornire meccanismi di cache per i database.
- Valuta l'architettura basata sugli eventi: l'utilizzo dell'architettura serverless consente inoltre di costruire un'architettura basata sugli eventi per applicazioni distribuite basate su microservizi, che aiuta a costruire soluzioni scalabili, resilienti, agili ed economiche.

Risorse

Documenti correlati:

- Calcolatore del costo totale di proprietà (TCO) di AWS
- AWS Serverless
- In cosa consiste l'architettura basata su eventi
- Classi di archiviazione di Amazon S3
- Prodotti cloud
- Amazon ElastiCache for Redis

Esempi correlati:

- Getting started with event-driven architecture
- Architettura basata su eventi

- How Statsig runs 100x more cost-effectively using Amazon ElastiCache for Redis
- Best practices for working with AWS Lambda functions

COST05-BP06 Esecuzione di un'analisi dei costi per diversi valori di utilizzo nel tempo

I carichi di lavoro possono cambiare nel corso del tempo. Alcuni servizi o funzionalità sono più convenienti a diversi livelli di utilizzo. Eseguendo l'analisi su ogni componente nel tempo e all'utilizzo previsto, il carico di lavoro rimane conveniente per tutta la sua durata.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

Quando AWS rilascia nuovi servizi e funzionalità, è possibile che i servizi ottimali per il carico di lavoro cambino. Tale cambiamento comporta un impegno, che dovrebbe essere commensurato ai vantaggi potenziali. La frequenza di revisione del carico di lavoro dipende dai requisiti dell'organizzazione. Se si tratta di un carico di lavoro con costi importanti, una rapida implementazione dei nuovi servizi massimizzerà i risparmi sui costi, e in tal caso una revisione più frequente può risultare vantaggiosa. Un altro stimolo importante per la revisione è il cambiamento dei modelli di utilizzo. Se si verificassero notevoli cambiamenti nell'utilizzo, ciò potrebbe indicare un maggiore vantaggio dei servizi alternativi.

Se si ha bisogno di trasferire i dati nel Cloud AWS è possibile scegliere i numerosi servizi offerti da AWS e gli strumenti dei partner per avere un supporto nella migrazione dei tuoi set di dati, sia che si tratti di file, database, immagini di macchine, volumi a blocchi o persino backup su nastro. Ad esempio, per spostare grandi quantità di dati da e verso AWS o elaborare dati in posizioni edge è possibile usare uno dei dispositivi AWS dedicati per migrare, in modo contenuto nei costi, petabyte di dati offline. Un altro esempio: per velocità di trasferimento dei dati più elevate, un servizio di connessione diretta può risultare più economico di una VPN e garantire la connettività coerente richiesta per la tua attività.

In base all'analisi dei costi per usi diversi nel tempo, rivedi le tue attività di dimensionamento. Analizza i risultati per vedere se la policy di dimensionamento può essere ottimizzata per aggiungere istanze con tipi di istanze e opzioni di acquisto diversi. Esamina le tue impostazioni per vedere se il minimo può essere ridotto per soddisfare le richieste degli utenti, ma con una dimensione inferiore del parco istanze, e aggiungi più risorse per i momenti attesi di incremento della domanda.

Esegui l'analisi dei costi per diversi utilizzi nel tempo discutendone con le parti interessate della tua organizzazione e usa la funzione di previsione di AWS Cost Explorer per prevedere l'impatto

potenziale di modifiche dei servizi. Monitora i trigger dei livelli di utilizzo con Budget AWS, gli allarmi di fatturazione di CloudWatch e AWS Cost Anomaly Detection per identificare e implementare in tempi rapidi i servizi più contenuti nei costi.

Passaggi dell'implementazione

- Definisci modelli di utilizzo previsti: collaborando con la tua organizzazione, ad esempio con i proprietari di prodotti e marketing, documenta quali sono i modelli di utilizzo previsti e attesi per il carico di lavoro. Discuti con le parti interessate dell'azienda dell'aumento dell'utilizzo e dei costi storici e previsti e verifica che tali incrementi siano in linea con i requisiti aziendali. Identifica i giorni, le settimane o i mesi di calendario in cui prevedi che un maggior numero di utenti userà le tue risorse AWS, il che indica che dovrai aumentare la capacità delle risorse esistenti o adottare servizi aggiuntivi per ridurre i costi e migliorare le performance.
- Esegui l'analisi dei costi in base all'utilizzo previsto: utilizzando i modelli di utilizzo definiti, esegui l'analisi in ciascuno di questi punti. Lo sforzo di analisi dovrebbe riflettere il potenziale risultato. Ad esempio, se la variazione dell'utilizzo è elevata, è necessario eseguire un'analisi accurata per verificare eventuali costi e modifiche. In altre parole, quando il costo aumenta dovrebbe aumentare anche l'utilizzo per l'azienda.

Risorse

Documenti correlati:

- Calcolatore del costo totale di proprietà (TCO) di AWS
- Classi di archiviazione di Amazon S3
- Prodotti cloud
- Amazon EC2 Auto Scaling
- Migrazione cloud dei dati
- AWS Snow Family

Video correlati:

- AWS OpsHub for Snow Family

COST 6. In che modo raggiungi gli obiettivi di costo quando selezioni il tipo, le dimensioni e il numero delle risorse?

Assicurati di scegliere la dimensione e il numero delle risorse appropriati per l'attività in questione. Selezionando il tipo, le dimensioni e il numero più convenienti, riduci al minimo gli sprechi.

Best practice

- COST06-BP01 Esecuzione della modellazione dei costi
- COST06-BP02 Selezione di tipo, dimensione e numero di risorse sulla base dei dati
- COST06-BP03 Selezione automatica del tipo e della dimensione della risorsa in base ai parametri
- COST06-BP04 Valutazione dell'utilizzo delle risorse condivise

COST06-BP01 Esecuzione della modellazione dei costi

Identifica i requisiti dell'organizzazione (come le esigenze aziendali e gli impegni esistenti) ed esegui la modellazione dei costi (costi complessivi) del carico di lavoro e di ciascuno dei suoi componenti. Esegui attività di analisi comparativa per il carico di lavoro in base ai diversi carichi previsti e confronta i costi. L'impegno di modellazione deve riflettere il potenziale risultato. Ad esempio, il tempo speso dovrebbe essere proporzionale al costo dei componenti.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

Esegui la modellazione dei costi per il tuo carico di lavoro e ciascuno dei suoi componenti per stabilire il giusto equilibrio tra le risorse e trova la dimensione appropriata per ogni risorsa nel carico di lavoro, sulla base di un determinato livello di prestazioni. La comprensione delle considerazioni sui costi può informare il business case dell'organizzazione e il processo decisionale quando si valutano i risultati di realizzazione del valore per l'implementazione del carico di lavoro pianificato.

Esegui attività di analisi comparativa per il carico di lavoro in base ai diversi carichi previsti e confronta i costi. L'impegno di modellazione deve riflettere il potenziale risultato. Ad esempio, il tempo speso dovrebbe essere proporzionale al costo dei componenti o ai risparmi previsti. Per le best practice, consulta Revisione della sezione del Principio dell'efficienza della performance di AWS Well-Architected Framework.

Come esempio, per creare la modellazione dei costi per un carico di lavoro con risorse di calcolo AWS Compute Optimizer può assistere con la modellazione dei costi per l'esecuzione dei carichi

di lavoro. Fornisce consigli di dimensionamento appropriato per le risorse di calcolo in base a una valutazione cronologica dell'utilizzo. Assicurati che gli agenti CloudWatch siano distribuiti sulle istanze Amazon EC2 per raccogliere le metriche della memoria che aiutano a fornire raccomandazioni più accurate all'interno di AWS Compute Optimizer. Questa è l'origine dati ideale per le risorse di calcolo perché è un servizio gratuito e utilizza il machine learning per generare più raccomandazioni a seconda dei livelli di rischio.

Esistono più servizi che è possibile utilizzare con log personalizzati come origini dati per le operazioni di ridimensionamento di altri servizi e componenti del carico di lavoro, ad esempio AWS Trusted Advisor, Amazon CloudWatch e Amazon CloudWatch Logs. AWS Trusted Advisor controlla le risorse e segnala quelle a basso utilizzo, che aiutano a dimensionare correttamente le risorse e a creare una modellazione dei costi.

Di seguito sono riportate le raccomandazioni per i parametri e i dati di modellazione dei costi:

- Il monitoraggio deve corrispondere in modo preciso all'esperienza degli utenti. Seleziona la granularità corretta per un dato periodo di tempo e scegli in modo ponderato il 99° percentile o quello massimo invece del valore medio.
- Seleziona la granularità corretta per il periodo di analisi richiesto per coprire tutti i cicli del carico di lavoro. Ad esempio, se esegui un'analisi di due settimane, potresti ignorare un ciclo mensile di utilizzo elevato, e questo potrebbe causare un provisioning insufficiente.
- Scegli i servizi AWS giusti per il carico di lavoro pianificato considerando gli impegni esistenti, i modelli di prezzo selezionati per altri carichi di lavoro e la capacità di innovare più rapidamente e di concentrarsi sul valore del core business.

Passaggi dell'implementazione

- Esegui una modellazione dei costi per le risorse: implementa il carico di lavoro o un proof of concept in un account separato con i tipi di risorse e dimensioni specifiche da testare. Esegui il carico di lavoro con i dati di test e registra i risultati di output, insieme ai dati relativi ai costi per il tempo in cui è stato eseguito il test. Quindi, implementa di nuovo il carico di lavoro o modifica i tipi e le dimensioni delle risorse ed esegui nuovamente il test. Includi i costi di licenza di qualsiasi prodotto che si possa utilizzare con queste risorse e i costi operativi stimati (manodopera o tecnici) per l'implementazione e la gestione di queste risorse durante la creazione di modelli di costo. Considera la modellazione dei costi per un periodo (orario, giornaliero, mensile, annuale o triennale).

Risorse

Documenti correlati:

- AWS Auto Scaling
- Identificare le opportunità per un dimensionamento corretto
- Funzionalità di Amazon CloudWatch
- Ottimizzazione dei costi: dimensionamento appropriato di Amazon EC2
- AWS Compute Optimizer
- Calcolatore dei prezzi AWS

Esempi correlati:

- Esegui una modellazione dei costi basata sui dati
- Stima il costo delle configurazioni di risorse AWS pianificate
- Scegli gli strumenti AWS corretti

COST06-BP02 Selezione di tipo, dimensione e numero di risorse sulla base dei dati

Seleziona la dimensione o il tipo di risorsa in base ai dati relativi al carico di lavoro e alle caratteristiche delle risorse come, ad esempio, elaborazione, memoria, throughput o scrittura intensiva. Questa selezione è tipicamente effettuata utilizzando una versione precedente (on-premise) del carico di lavoro, utilizzando la documentazione o altre fonti di informazione sul carico di lavoro.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Amazon EC2 offre un'ampia selezione di tipi di istanza con vari livelli di CPU, memoria, archiviazione e capacità di rete per adattarsi a diversi casi d'uso. Questi tipi di istanza offrono diverse combinazioni di CPU, memoria, archiviazione e funzionalità di rete, che garantiscono versatilità nella scelta delle risorse giuste per i tuoi progetti. Ogni tipo di istanza è disponibile in più dimensioni, per consentire di adattare le risorse alle richieste del carico di lavoro. Per determinare il tipo di istanza necessario, acquisisci i dettagli sui requisiti di sistema dell'applicazione o del software che intendi eseguire sull'istanza. Tali dettagli devono includere le informazioni seguenti:

- Sistema operativo

- Numero di core della CPU
- Core della GPU
- Quantità di memoria di sistema (RAM)
- Tipo e spazio di archiviazione
- Requisiti di larghezza di banda della rete

Identifica lo scopo dei requisiti di calcolo e l'istanza necessaria, quindi analizza le varie famiglie di istanze Amazon EC2. Amazon offre le seguenti famiglie di tipi di istanza:

- Per uso generico
- Ottimizzate per il calcolo
- Ottimizzate per la memoria
- Ottimizzate per l'archiviazione
- Calcolo accelerato
- Ottimizzate per il calcolo ad alte prestazioni (HPC)

Per una comprensione più approfondita degli scopi e dei casi d'uso specifici che una particolare famiglia di istanze Amazon EC2 può soddisfare, consulta Instance typesAWS.

L'acquisizione dei requisiti di sistema è fondamentale per selezionare la famiglia e il tipo di istanze specifici più adatti alle proprie esigenze. I nomi dei tipi di istanza sono composti dal nome della famiglia e dalla dimensione dell'istanza. Ad esempio, l'istanza t2.micro appartiene alla famiglia T2 ed è di dimensioni ridotte.

Seleziona la dimensione o il tipo di risorsa in base al carico di lavoro e alle caratteristiche delle risorse come, ad esempio, calcolo, memoria, velocità di trasmissione effettiva o uso intensivo di operazioni di scrittura. Questa selezione è in genere effettuata ricorrendo alla modellazione dei costi, a una versione precedente del carico di lavoro (ad esempio una versione on-premise), alla documentazione o ad altre fonti di informazione sul carico di lavoro (come whitepaper o soluzioni pubblicate). L'uso di calcolatori dei prezzi AWS o di strumenti di gestione dei costi può aiutare a elaborare decisioni informate su tipi, dimensioni e configurazioni delle istanze.

Passaggi dell'implementazione

- Seleziona le risorse in base ai dati: utilizza i dati di modellazione dei costi per selezionare il livello di utilizzo previsto del carico di lavoro e scegliere il tipo e la dimensione delle risorse specificati.

Basandoti sui dati di modellazione dei costi, determina il numero di CPU virtuali, la memoria totale (GiB), il volume dell'archivio dell'istanza locale (GB), i volumi Amazon EBS e il livello di prestazioni della rete, tenendo conto della velocità di trasferimento dei dati richiesta per l'istanza. Effettua sempre selezioni basate su analisi dettagliate e dati accurati per ottimizzare le prestazioni e contemporaneamente gestire i costi in modo efficace.

Risorse

Documenti correlati:

- Tipi di istanza AWS
- AWS Auto Scaling
- Funzionalità di Amazon CloudWatch
- Cost Optimization: EC2 Right Sizing

Video correlati:

- Selecting the right Amazon EC2 instance for your workloads
- Right size your service

Esempi correlati:

- It just got easier to discover and compare Amazon EC2 instance types

COST06-BP03 Selezione automatica del tipo e della dimensione della risorsa in base ai parametri

Utilizza i parametri del carico di lavoro in esecuzione per selezionare la dimensione e il tipo giusti per ottimizzare i costi. Esegui il provisioning in modo appropriato di throughput, dimensione e spazio di archiviazione per servizi di calcolo, memorizzazione, gestione dati e di rete. Questa operazione può essere eseguita con un ciclo di feedback, ad esempio attraverso l'auto scaling o tramite codice personalizzato nel carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Crea un ciclo di feedback all'interno del carico di lavoro che utilizza i parametri attivi del carico di lavoro in esecuzione per apportarvi modifiche. È possibile utilizzare un servizio gestito come AWS

Auto Scaling che può essere configurato per eseguire le giuste operazioni di dimensionamento per conto proprio. AWS fornisce anche API, SDK e funzionalità che permettono alle risorse di essere modificate con un minimo sforzo. Puoi programmare un carico di lavoro affinché arresti e riavvii un'istanza Amazon EC2 per consentire una modifica delle dimensioni o del tipo di istanza. Ciò offre i vantaggi del dimensionamento appropriato, eliminando al contempo quasi tutti i costi operativi necessari per apportare la modifica.

Alcuni servizi AWS includono una selezione integrata automatica di tipo o dimensione come Amazon Simple Storage Service Intelligent-Tiering. Basandosi sui modelli di utilizzo, Amazon S3 Intelligent-Tiering sposta automaticamente i dati tra due livelli di accesso: frequente e poco frequente.

Passaggi dell'implementazione

- Incrementa l'osservabilità configurando i parametri del carico di lavoro: acquisisci i parametri chiave del carico di lavoro. Questi parametri, come ad esempio l'output del carico di lavoro, forniscono un'indicazione dell'esperienza del cliente e danno indicazioni legate alle differenze tra tipi e dimensioni di risorse, come l'utilizzo di CPU e memoria. Per le risorse di calcolo, analizza i dati sulle prestazioni per dimensionare correttamente le istanze Amazon EC2. Identifica le istanze inattive e quelle sottoutilizzate. Le metriche chiave da controllare sono l'utilizzo della CPU e della memoria (ad esempio, il 40% di utilizzo della CPU per il 90% del tempo, come spiegato in Ridimensionamento con AWS Compute Optimizer e utilizzo della memoria abilitati). Identifica le istanze con un utilizzo massimo della CPU e della memoria inferiore al 40% su un periodo di quattro settimane. Questi sono i casi in cui è necessario dimensionare correttamente il sistema per ridurre i costi. Per le risorse di archiviazione come Amazon S3 è possibile utilizzare Amazon S3 Storage Lens che, per impostazione predefinita, consente di visualizzare 28 parametri in varie categorie a livello di bucket e 14 giorni di dati storici nel pannello di controllo. Per analizzare specifici parametri, si possono applicare dei filtri su riepilogo e ottimizzazione dei costi o eventi all'interno del pannello di controllo di Amazon S3 Storage Lens.
- Visualizza le raccomandazioni per il dimensionamento appropriato: utilizza le raccomandazioni per il dimensionamento appropriato in AWS Compute Optimizer e lo strumento per il dimensionamento appropriato di Amazon EC2 nella console di gestione dei costi, oppure esamina l'attività di dimensionamento appropriato di AWS Trusted Advisor per apportare le opportune regolazioni sul tuo carico di lavoro. Quando si dimensionano in modo appropriato diverse risorse è importante usare gli strumenti giusti e seguire le linee guida al dimensionamento appropriato che si tratti di un'istanza Amazon EC2, di classi di archiviazione AWS o di tipi di istanza Amazon RDS. Per le risorse di archiviazione è possibile utilizzare Amazon S3 Storage Lens, che offre visibilità sull'utilizzo dello spazio di archiviazione di oggetti e sulle tendenze delle attività e fornisce raccomandazioni operative per ottimizzare i costi e applicare le best practice di protezione dei

dati. Utilizzando le raccomandazioni contestuali che Amazon S3 Storage Lens deriva dall'analisi dei parametri all'interno della tua organizzazione, si possono adottare misure immediate per ottimizzare lo spazio di archiviazione.

- Seleziona automaticamente il tipo e la dimensione delle risorse in base ai parametri: utilizzando i parametri del carico di lavoro, seleziona manualmente o automaticamente le relative risorse. Per le risorse di calcolo, la configurazione di AWS Auto Scaling o l'implementazione di codice all'interno dell'applicazione può ridurre lo sforzo necessario in caso di modifiche frequenti e permettere di implementare potenzialmente eventuali modifiche più velocemente rispetto a un processo manuale. Si può lanciare e scalare automaticamente un parco istanze on demand e di istanze Spot all'interno di un singolo gruppo Auto Scaling. Oltre a ricevere sconti per l'utilizzo di Istanze Spot, è possibile utilizzare Istanze Riservate o Savings Plans per ricevere tariffe scontate rispetto al normale prezzo delle Istanze on demand. La combinazione di tutti questi fattori consente di ottimizzare i risparmi sui costi delle istanze Amazon EC2 e di determinare il dimensionamento e le prestazioni desiderate per la tua applicazione. Si può anche usare una strategia di selezione del tipo di istanza basata su attributi (ABS) nei gruppi Auto Scaling (ASG) che consenta di esprimere i requisiti dell'istanza come un set di attributi, ad esempio vCPU, memoria e spazio di archiviazione. È possibile utilizzare automaticamente i tipi di istanza di nuova generazione quando vengono rilasciati e accedere a una gamma più ampia di capacità con le istanze Spot di Amazon EC2. Il parco istanze Amazon EC2 e Amazon EC2 Auto Scaling selezionano e avviano istanze che corrispondono agli attributi specificati, eliminando la necessità di scegliere manualmente i tipi di istanza. Per le risorse di archiviazione puoi usare le funzionalità di Intelligent Tiering di Amazon S3 e accesso non frequente di Amazon EFS, che consentono di selezionare automaticamente le classi di archiviazione che offrono risparmi automatici sui relativi costi quando cambiano i modelli di accesso ai dati, senza impatto sulle prestazioni né sovraccarico operativo.

Risorse

Documenti correlati:

- AWS Auto Scaling
- Dimensionamento appropriato di AWS
- AWS Compute Optimizer
- Funzionalità di Amazon CloudWatch
- Configurazione di CloudWatch
- Pubblicazione di parametri personalizzati in CloudWatch
- Nozioni di base su Amazon EC2 Auto Scaling

- Amazon S3 Storage Lens
- Intelligent Tiering di Amazon S3
- Accesso non frequente di Amazon EFS
- Avvia un'istanza Amazon EC2 utilizzando l'SDK

Video correlati:

- Dimensiona in modo appropriato i tuoi servizi

Esempi correlati:

- Selezione dell'istanza basata sugli attributi per Auto Scaling per il parco istanze Amazon EC2
- Ottimizzazione dei costi di Amazon Elastic Container Service utilizzando il dimensionamento programmato
- Dimensionamento predittivo con Amazon EC2 Auto Scaling
- Ottimizza i costi e acquista visibilità sull'utilizzo con Amazon S3 Storage Lens
- Well-Architected Labs: raccomandazioni per il dimensionamento appropriato (Livello 100)
- Well-Architected Labs: dimensionamento appropriato con AWS Compute Optimizer e l'utilizzo della memoria abilitati (Livello 200)

COST06-BP04 Valutazione dell'utilizzo delle risorse condivise

Per i servizi già implementati a livello di organizzazione per più business unit, valuta l'uso delle risorse condivise per aumentare l'utilizzo e ridurre il costo totale di proprietà (TCO). L'utilizzo delle risorse condivise può essere un'opzione conveniente per centralizzare la gestione e i costi usando le soluzioni esistenti, condividendo i componenti o in entrambi i casi. Gestisci le funzioni comuni, come monitoraggio, backup e connettività, entro il limite dell'account o in un account dedicato. Inoltre, puoi diminuire i costi implementando la standardizzazione nonché riducendo la duplicazione e la complessità.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Se più carichi di lavoro eseguono la stessa funzione, utilizza le soluzioni esistenti e i componenti condivisi per migliorare la gestione e ottimizzare i costi. Prendi in considerazione l'utilizzo delle

risorse esistenti, in particolare quelle condivise, come server di database non di produzione o servizi di directory, per contenere i costi del cloud seguendo le best practice di sicurezza e le normative dell'organizzazione. Per realizzare valore ed efficienza ottimali, è fondamentale utilizzare i report di showback e i meccanismi di chargeback per riallocare i costi alle aree pertinenti dell'azienda che determinano i consumi.

I report di showback suddividono i costi del cloud in categorie attribuibili, come consumatori, business unit, conti di contabilità generale o altre entità responsabili. L'obiettivo dei report di showback è mostrare a team, business unit o singole persone il costo delle risorse cloud consumate.

Il chargeback alloca la spesa del servizio centrale alle unità di costo sulla base di una strategia definita per uno specifico processo di gestione finanziaria. Per i clienti, il chargeback addebita il costo sostenuto da un account di servizi condivisi a diverse categorie di costi finanziari definite per un processo di report dei clienti. Stabilendo i meccanismi di chargeback, puoi dichiarare i costi sostenuti da diverse business unit, prodotti e team.

I carichi di lavoro possono essere classificati come critici e non critici. Sulla base di questa classificazione, utilizza le risorse condivise con configurazioni generali per i carichi di lavoro meno critici. Per ottimizzare ulteriormente i costi, usa i server dedicati esclusivamente per i carichi di lavoro critici. Condividi o alloca le risorse in più account per gestirle in modo efficiente. La condivisione è sicura e non compromette la struttura organizzativa anche quando gli ambienti di sviluppo, test e produzione sono separati.

Per migliorare la comprensione e ottimizzare i costi e l'utilizzo delle applicazioni containerizzate, utilizza i dati di allocazione dei costi suddivisi che consentono di allocare i costi alle singole entità aziendali in base al modo in cui l'applicazione consuma le risorse di calcolo e memoria condivise. I dati di allocazione dei costi suddivisi consentono di utilizzare showback e chargeback a livello di attività nei carichi di lavoro dei container in esecuzione su Amazon Elastic Container Service (Amazon ECS) o Amazon Elastic Kubernetes Service (Amazon EKS).

Per le architetture distribuite, crea un VPC di servizi condivisi che fornisca l'accesso centralizzato ai servizi condivisi richiesti dai carichi di lavoro in ogni VPC. Questi servizi condivisi possono includere risorse quali servizi di directory o endpoint VPC. Per ridurre il sovraccarico e i costi amministrativi, condividi le risorse da una posizione centrale invece di crearle in ogni VPC.

Quando si utilizzano le risorse condivise, è possibile ridurre i costi operativi, massimizzare l'utilizzo delle risorse e migliorare la coerenza. In una progettazione multi-account, puoi risparmiare sui costi ospitando alcuni servizi AWS centralmente e accedendovi tramite diverse applicazioni e account in un hub. Puoi utilizzare AWS Resource Access Manager (AWS RAM) per condividere altre risorse

comuni, quali sottoreti VPC e collegamenti AWS Transit Gateway, AWS Network Firewall o pipeline Amazon SageMaker. In un ambiente multi-account, usa AWS RAM per creare una risorsa una sola volta e condividerla con altri account.

Le organizzazioni devono applicare i tag in modo efficace ai costi condivisi e verificare che non vi siano parti significative dei costi senza tag o allocazione. Se non si allocano i costi condivisi in modo efficace e nessuno se ne assume la responsabilità della gestione, i costi condivisi del cloud possono aumentare vertiginosamente. È necessario sapere dove sostieni i costi a livello di risorse, carico di lavoro, team oppure organizzazione perché questa conoscenza migliora la tua comprensione del valore fornito al livello applicabile rispetto ai risultati aziendali raggiunti. In definitiva, le organizzazioni ottengono il vantaggio del risparmio sui costi grazie alla condivisione dell'infrastruttura cloud. Incoraggia l'allocazione dei costi sulle risorse condivise del cloud per ottimizzare la spesa del cloud.

Passaggi dell'implementazione

- Valuta le risorse esistenti: esamina i carichi di lavoro esistenti che utilizzano servizi simili per il carico di lavoro. A seconda dei componenti del carico di lavoro, considera le piattaforme esistenti, se la logica aziendale o i requisiti tecnici lo consentono.
- Usa la condivisione delle risorse in AWS RAM e limita di conseguenza: utilizza AWS RAM per condividere le risorse con altri account AWS all'interno della tua organizzazione. Con la condivisione non dovrai duplicare le risorse in più account e riduci al minimo l'onere operativo della manutenzione delle risorse. Questo processo ti consente inoltre di condividere in modo sicuro le risorse create con i ruoli e gli utenti del proprio account e di altri Account AWS.
- Applica i tag alle risorse: definisci i tag delle risorse utilizzate per i report dei costi e classifica le risorse all'interno delle categorie di costo. Attiva questi tag delle risorse relativi ai costi per l'allocazione dei costi per ottenere visibilità sull'utilizzo delle risorse AWS. Concentrati sulla creazione di un livello adeguato di granularità rispetto alla visibilità dei costi e dell'utilizzo e influenza i comportamenti di consumo del cloud attraverso la creazione di report sull'allocazione dei costi e il monitoraggio dei KPI.

Risorse

Best practice correlate:

- SEC03-BP08 Condivisione delle risorse in modo sicuro all'interno dell'organizzazione

Documenti correlati:

- What is AWS Resource Access Manager?
- AWS services that you can use with AWS Organizations
- Shareable AWS resources
- AWS Cost and Usage (CUR) Queries

Video correlati:

- AWS Resource Access Manager - Granular access control with managed permissions
- How to design your AWS cost allocation strategy
- AWS Cost Categories

Esempi correlati:

- How-to chargeback shared services: An AWS Transit Gateway example
- How to build a chargeback/showback model for Savings Plans using the CUR
- Using VPC Sharing for a Cost-Effective Multi-Account Microservice Architecture
- Improve cost visibility of Amazon EKS with AWS Split Cost Allocation Data
- Improve cost visibility of Amazon ECS and AWS Batch with AWS Split Cost Allocation Data

COST 7. In che modo impieghi i modelli di prezzo per ridurre i costi?

Utilizza il modello di prezzo più appropriato per le tue risorse per ridurre al minimo le spese.

Best practice

- COST07-BP01 Esecuzione di un'analisi del modello di prezzo
- COST07-BP02 Scegli le regioni in base al costo
- COST07-BP03 Selezione di contratti di terze parti con condizioni economicamente convenienti
- COST07-BP04 Implementazione di modelli di determinazione dei prezzi per tutti i componenti del carico di lavoro
- COST07-BP05 Esecuzione dell'analisi del modello di prezzo a livello di account di gestione

COST07-BP01 Esecuzione di un'analisi del modello di prezzo

Analizza ogni componente del carico di lavoro. Determina se il componente e le risorse saranno in esecuzione per periodi prolungati (per sconti a fronte di impegni) o dinamici e di breve durata (per spot oppure on demand). Esegui un'analisi sul carico di lavoro utilizzando i suggerimenti presenti negli strumenti di gestione dei costi e applica le regole aziendali a tali suggerimenti per ottenere rendimenti elevati.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

AWS ha più modelli di prezzo che consentono di pagare per le risorse nel modo più conveniente e adatto alle esigenze della tua organizzazione e in base al prodotto. Lavora con i tuoi team per stabilire il modello di prezzi più appropriato. Spesso il modello di prezzi è costituito da più opzioni, in base alla tua disponibilità

Istanze on demand consentono di pagare capacità di elaborazione o di database all'ora o al secondo (minimo 60 secondi) in base a quali istanze esegui, senza impegni nel lungo termine o pagamenti anticipati.

Savings Plans sono un modello di prezzi flessibile che offre prezzi contenuti sull'utilizzo di Amazon EC2, Lambda e AWS Fargate (Fargate), in cambio di un impegno per un uso sostenuto (misurato in dollari per ora) in un periodo di un anno o di tre anni.

Istanze spot sono un meccanismo di prezzo Amazon EC2 che consente di richiedere capacità di elaborazione inutilizzata a una tariffa oraria scontata (fino al 90% rispetto al prezzo on-demand) senza un impegno iniziale.

Istanze riservate consentono di ottenere uno sconto fino al 75% con un pagamento anticipato per la capacità. Per maggiori dettagli consulta Ottimizzare i costi con le prenotazioni.

Potresti scegliere di includere un Savings Plan per le risorse associate alla produzione, alla qualità e agli ambienti di sviluppo. In alternativa, poiché le risorse dell'ambiente di sperimentazione (sandbox) vengono attivate solo se necessarie, potresti adottare un modello on-demand per le risorse presenti in quel contesto. Use le Istanze spot Amazon per ridurre i costi Amazon EC2 oppure usa Compute Savings Plans per ridurre i costi Amazon EC2, Fargate e Lambda. Lo strumento di suggerimenti AWS Cost Explorer offre opportunità di ottenere sconti con i Saving Plan.

Se hai acquistato Istanze riservate per Amazon EC2 in passato o hai definito procedure di allocazione dei costi all'interno della tua organizzazione, puoi continuare a usare le Istanze riservate

Amazon EC2 per il momento. Tuttavia, ti consigliamo di lavorare su una strategia per usare Savings Plans in futuro come meccanismo più flessibile di risparmio sui costi. Puoi aggiornare i suggerimenti Savings Plans (SP) in AWS Cost Management per generare nuovi suggerimenti di Saving Plan in qualsiasi momento. Usa le Istanze riservate (RI) per ridurre i costi Amazon RDS, Amazon Redshift, Amazon ElastiCache e Amazon OpenSearch Service. Saving Plan e Istanze riservate sono disponibili in tre opzioni: pagamento anticipato totale, pagamento anticipato parziale e nessun pagamento anticipato. Usa le raccomandazioni fornite nei consigli di acquisto RI e SP AWS Cost Explorer.

Per trovare opportunità per i carichi di lavoro Spot, utilizza una visualizzazione oraria dell'utilizzo complessivo e cerca periodi regolari di variazione dell'utilizzo o di elasticità. Puoi usare le Istanze spot per diverse applicazioni flessibili e tolleranti ai guasti. Tra gli esempi figurano server Web stateless, endpoint di API, applicazioni di big data e analisi, carichi di lavoro containerizzati, CI/CD e altri carichi di lavoro flessibili.

Analizza se le tue istanze Amazon EC2 e Amazon RDS possono essere disattivate quando non le usi (dopo l'orario di lavoro e nei weekend). In questo modo potrai ridurre i costi di almeno il 70% rispetto al loro utilizzo 24 ore su 24, 7 giorni su 7. Se hai cluster Amazon Redshift che devono essere disponibili solo in orari specifici, puoi metterli in pausa e poi riattivarli. Quando il cluster Amazon Redshift o Amazon EC2 e l'istanza Amazon RDS vengono arrestati, la fattura relativa all'elaborazione si arresta e si applicano solo i costi di archiviazione.

Da notare che le Prenotazioni della capacità on-demand (ODCR) non sono uno sconto sul prezzo. Le prenotazioni della capacità vengono addebitate alla tariffa on-demand equivalente, sia che tu esegua istanze con capacità riservata oppure no. Tali prenotazioni devono essere prese in considerazione quando hai bisogno di offrire capacità sufficiente alle risorse che desideri eseguire. Le ODCR non devono essere considerate un impegno nel lungo termine, poiché possono essere annullate quando non ne hai più bisogno, ma possono anche approfittare degli sconti offerti da Savings Plans o dalle Istanze riservate.

Passaggi dell'implementazione

- Analizza l'elasticità del carico di lavoro: Utilizzando la granularità oraria in Cost Explorer o un pannello di controllo personalizzato, analizza l'elasticità del tuo carico di lavoro. Vai alla ricerca di modifiche regolari del numero di istanze in esecuzione. Le istanze in esecuzione per brevi periodi di tempo sono candidate per essere istanze Spot o serie di istanze Spot.
  - Well-Architected Lab: Cost Explorer
  - Well-Architected Labs: visualizzazione dei costi

- Esamina i contratti esistenti sui prezzi: esamina i contratti o gli impegni in essere per le esigenze nel lungo termine. Analizza ciò di cui disponi ora e fino a che punto gli impegni presi vengono sfruttati. Sfrutta sconti contrattuali preesistenti o accordi aziendali. Gli Accordi aziendali offrono ai clienti la possibilità di personalizzare i contratti in modo che siano rispondenti alle esigenze aziendali. Per accordi nel lungo termine, prendi in considerazione gli sconti dei prezzi riservati, le Istanze riservate o Savings Plans per il tipo di istanza specifico, la famiglia delle istanze, Regione AWS e le zone di disponibilità.
- Esegui un'analisi degli sconti in seguito a un impegno contrattuale: tramite l'uso di Cost Explorer nel tuo account, esamina Savings Plans e i suggerimenti delle Istanze riservate. Per verificare di implementare le raccomandazioni corrette con gli sconti e i rischi richiesti, segui i Well-Architected Labs.

Risorse

Documenti correlati:

- Accesso alle raccomandazioni di istanza riservata
- Opzioni di acquisto delle istanze
- AWS Enterprise

Video correlati:

- Risparmia fino al 90% ed esegui i carichi di lavoro di produzione su Spot

Esempi correlati:

- Well-Architected Lab: Cost Explorer
- Well-Architected Labs: visualizzazione dei costi
- Well-Architected Lab: modelli di prezzo

COST07-BP02 Scegli le regioni in base al costo

La determinazione dei prezzi delle risorse può essere diversa in ciascuna regione. Individua le differenze di costo a livello regionale ed esegui la distribuzione solo nelle Regioni con costi più elevati per soddisfare i requisiti di latenza, posizionamento fisico dei dati e sovranità dei dati. La

considerazione del costo della regione garantisce il pagamento del prezzo complessivo più basso per questo carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

L' infrastruttura Cloud AWS è globale, ospitata in più sedi in tutto il mondoe costruita in base a Regioni AWS, zone di disponibilità, zone locali, AWS Outposts e zone di lunghezza d'onda. Una regione è una posizione fisica nel mondo e ogni regione è un'area geografica separata in cui AWS ha più zone di disponibilità. Le zone di disponibilità, che sono più sedi isolate all'interno di ogni regione, sono costituite da uno o più data center discreti, ciascuno con alimentazione, rete e connettività ridondanti.

Ogni Regione AWS opera nelle condizioni di mercato locale e il prezzo delle risorse è diverso in ogni Regione, ad esempio a causa delle differenze nel costo della terra, della fibra, dell'elettricità e delle tasse. Scegli una regione specifica per gestire un componente o tutta la tua soluzione in modo da eseguirla al minor prezzo possibile a livello globale. Utilizza lo strumento Calcolatore dei prezzi AWS per stimare i costi del carico di lavoro in varie Regioni, cercando i servizi per tipo di località (Regione, zona di lunghezza d'onda e zona locale) e Regione.

Quando progetti le tue soluzioni, una best practice da seguire è quella di cercare di posizionare le risorse di calcolo vicino agli utenti per offrire una latenza inferiore e una forte sovranità dei dati. Seleziona la posizione geografica in base alle esigenze di business, privacy dei dati, performance e requisiti di sicurezza. Per le applicazioni con utenti finali globali, utilizza più sedi.

Utilizza le regioni che offrono prezzi più bassi per i servizi AWS per distribuire i carichi di lavoro se non hai obblighi in materia di privacy dei dati, sicurezza e requisiti aziendali. Ad esempio, se la regione predefinita è ap-southeasth-2 (Sydney) e se non ci sono restrizioni (privacy dei dati, sicurezza, ad esempio) per l'utilizzo di altre regioni, l'implementazione di istanze Amazon EC2 non critiche (sviluppo e test) nella regione north-east-1 (N. Virginia) costerà meno.

	Conformità	Latenza	Costo	Servizi/Caratteristiche
Regione 1	✓	15 ms	$$	✓
Regione 2	✓	20 ms	$$$	X
Regione 3	✓	80 ms	$	✓
Regione 4	✓	15 ms	$$	✓
Regione 5	✓	20 ms	$$$	X
Regione 6	✓	15 ms	$	✓
Regione 7	✓	80 ms	$	✓
Regione 8	✓	15 ms	$	X

Tabella della matrice delle caratteristiche della Regione

La tabella a matrice precedente mostra che la Regione 4 è l'opzione migliore per questo scenario specifico perché la latenza è bassa rispetto ad altre Regioni, il servizio è disponibile ed è la Regione meno costosa.

Passaggi dell'implementazione

- Rivedi i prezzi della Regione AWS: analizza i costi del carico di lavoro nella regione corrente. A partire dai costi più elevati per servizio e tipo di utilizzo, calcola i costi in altre regioni disponibili. Se il risparmio previsto supera il costo di spostamento del componente o del carico di lavoro, esegui la migrazione alla nuova regione.
- Rivedi i requisiti per implementazioni multi-regione: analizza i requisiti e gli obblighi aziendali (privacy dei dati, sicurezza o prestazioni) per scoprire se ci sono restrizioni che impediscono di utilizzare più Regioni. Se non ci sono obblighi che limitano l'utilizzo di una sola regione, allora utilizza più regioni.
- Analizza il trasferimento di dati richiesto: considera i costi per il trasferimento dei dati quando selezioni le Regioni. Mantieni i dati vicino ai clienti e alle risorse. Seleziona le Regioni AWS meno costose in cui confluiscono i dati e che richiedono trasferimenti minimi di dati. A seconda dei requisiti aziendali per il trasferimento dei dati, puoi utilizzare Amazon CloudFront, AWS PrivateLink,

AWS Direct Connecte AWS Virtual Private Network per ridurre i costi di rete, nonché migliorare le prestazioni e la sicurezza.

Risorse

Documenti correlati:

- Accesso alle raccomandazioni di istanza riservata
- Prezzi di Amazon EC2
- Opzioni di acquisto dell'istanza
- Tabella delle regioni

Video correlati:

- Risparmia fino al 90% ed esegui i carichi di lavoro di produzione su Spot

Esempi correlati:

- Panoramica dei costi di trasferimento dei dati per architetture comuni
- Considerazioni sui costi per implementazioni globali
- elementi da considerare quando si seleziona una regione per i propri carichi di lavoro
- Well-Architected Labs: limita l'utilizzo dei servizi per Regione (Level 200)

COST07-BP03 Selezione di contratti di terze parti con condizioni economicamente convenienti

Gli accordi e i termini convenienti assicurano che i costi di questi servizi siano ridimensionati in base ai vantaggi che offrono. Seleziona gli accordi e i prezzi che si ridimensionano quando forniscono ulteriori vantaggi alla tua organizzazione.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Sul mercato esistono diversi prodotti che possono aiutarti a gestire i costi negli ambienti cloud. In termini di funzionalità possono presentare alcune differenze che dipendono dalle esigenze del cliente, ad esempio alcuni clienti sono più interessati alla governance o alla visibilità dei costi mentre

altri all'ottimizzazione di questi ultimi. Un fattore chiave per rendere più efficaci l'ottimizzazione e la governance dei costi è l'utilizzo dello strumento giusto con le funzionalità necessarie combinato al giusto modello di prezzo. Questi prodotti hanno modelli di prezzo diversi. Alcuni addebitano una determinata percentuale dell'importo fatturato mensilmente, mentre altri addebitano una percentuale dei risparmi realizzati. Idealmente, dovresti pagare solo ciò che hai effettivamente utilizzato.

Quando utilizzi soluzioni o servizi di terze parti nel cloud, è importante che le strutture dei prezzi siano allineate ai risultati desiderati. I prezzi devono essere adattati in base ai risultati e al valore che forniscono. Ad esempio, se utilizzi un software che contempla una percentuale del risparmio che fornisce, più risparmi (come risultato) e maggiore sarà l'importo addebitato. I contratti di licenza in cui paghi di più all'aumentare delle spese potrebbero non essere sempre nel tuo interesse ai fini dell'ottimizzazione dei costi. Tuttavia, se il fornitore offre vantaggi evidenti per tutte le voci incluse in fattura, questa tariffa scalare potrebbe essere giustificata.

Ad esempio, una soluzione che fornisce suggerimenti per Amazon EC2 e addebita una percentuale dell'intera fattura può diventare più dispendiosa se utilizzi altri servizi che non procurano alcun vantaggio. Un altro esempio è un servizio gestito che viene addebitato a una percentuale del costo delle risorse gestite. Una dimensione di istanza più grande potrebbe non richiedere necessariamente un maggiore impegno di gestione, ma potrebbe comportare un addebito superiore. Verifica che queste disposizioni tariffarie dei servizi includano un programma di ottimizzazione dei costi o funzionalità di servizio volte a migliorare l'efficienza.

I clienti potrebbero trovare i prodotti sul mercato più avanzati o più facili da usare. È necessario considerare il costo di questi prodotti e valutare i potenziali risultati di ottimizzazione dei costi a lungo termine.

Passaggi dell'implementazione

- Analizza i contratti e le condizioni stabilite con le terze parti: esamina i prezzi nei contratti di terze parti. Esegui la modellazione per diversi livelli di utilizzo e considera i nuovi costi, come il nuovo utilizzo del servizio o aumenti dei servizi attuali a causa della crescita del carico di lavoro. Decidi se i costi aggiuntivi forniscono i vantaggi necessari alla tua azienda.

Risorse

Documenti correlati:

- Accesso alle raccomandazioni di istanza riservata
- Opzioni di acquisto delle istanze

Video correlati:

- Risparmia fino al 90% ed esegui i carichi di lavoro di produzione su Spot

COST07-BP04 Implementazione di modelli di determinazione dei prezzi per tutti i componenti del carico di lavoro

Le risorse in esecuzione in modo permanente devono utilizzare la capacità riservata, ad esempio Savings Plans o istanze riservate. La capacità a breve termine è configurata per usare le istanze Spot o la serie di istanze Spot. Le istanze on demand vengono utilizzate solo per carichi di lavoro a breve termine che non possono essere interrotti e che non durano abbastanza a lungo per la capacità riservata, tra il 25% e il 75% del periodo, a seconda del tipo di risorsa.

Livello di rischio associato alla mancata adozione di questa best practice: basso

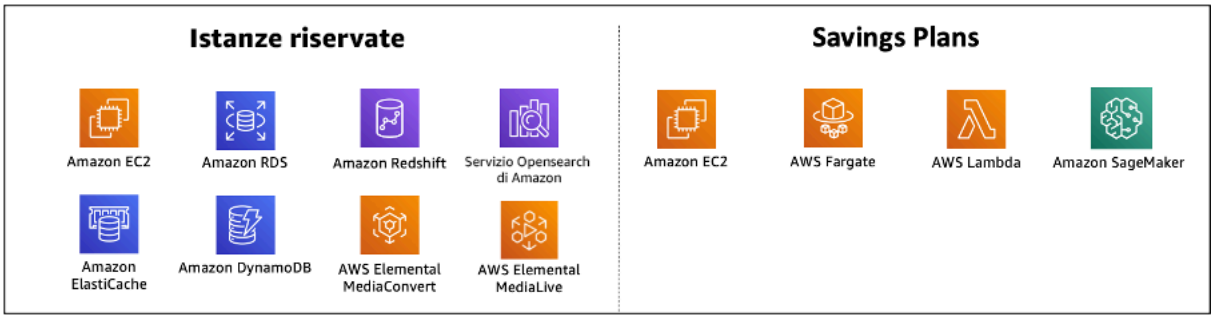
Guida all'implementazione

Per migliorare l'efficienza in termini di costi, AWS fornisce diversi consigli sull'impegno economico basati sull'utilizzo pregresso. Puoi utilizzare questi consigli per capire cosa puoi risparmiare e il livello di impegno richiesto. Puoi utilizzare questi servizi come istanze on demand o istanze spot oppure impegnarti per un determinato periodo di tempo e ridurre i costi delle istanze on demand mediante istanze riservate (RI) e Savings Plans (Savings Plans). Per ottimizzare il carico di lavoro, è necessario comprendere non solo i singoli componenti del carico di lavoro e i vari servizi AWS, ma anche gli sconti applicati agli impegni, le opzioni di acquisto e le istanze spot per questi servizi.

Considera i requisiti dei componenti del tuo carico di lavoro e valuta i diversi modelli di prezzo per questi servizi. Definisci il requisito di disponibilità dei componenti. Determina se ci sono più risorse indipendenti che eseguono la funzione nel carico di lavoro e quali sono i requisiti del carico di lavoro nel corso del tempo. Confronta il costo delle risorse utilizzando il modello di prezzo on demand predefinito e altri modelli applicabili. Tieni conto di qualsiasi potenziale cambiamento nelle risorse o nei componenti del carico di lavoro.

Analizza, ad esempio, questa architettura di applicazione Web su AWS. Questo carico di lavoro di esempio è composto da più servizi AWS, come Amazon Route 53, AWS WAF, Amazon CloudFront, istanze Amazon EC2, istanze Amazon RDS, sistemi di bilanciamento del carico, archiviazione Amazon S3 e Amazon Elastic File System (Amazon EFS). È necessario esaminare ciascuno di questi servizi e individuare le potenziali opportunità di risparmio sui costi con diversi modelli di prezzo. Alcuni potrebbero essere idonei per le istanze riservate (RI) o per il modello Savings Plans, mentre altri potrebbero essere disponibili solo nelle istanze on demand. Come illustrato nell'immagine

seguente, alcuni servizi AWS possono essere eseguiti utilizzando le istanze riservate (RI) o il modello Savings Plans.



Servizi AWS impegnati utilizzando istanze riservate e Savings Plans

Passaggi dell'implementazione

- Implementazione dei modelli di prezzo: utilizza i risultati dell'analisi, acquista Savings Plans, istanze riservate (RI) o implementa istanze spot. Se è il tuo primo acquisto a fronte di impegni, scegli i primi cinque o dieci consigli nell'elenco, quindi monitora e analizza i risultati nel corso del mese successivo o dei due mesi successivi. AWS Cost Management Console ti guida durante l'intero processo. Rivedi i consigli relativi all'istanza riservata (RI) o al modello Savings Plans sulla console, personalizza i consigli (tipo, pagamento e durata) e rivedi l'impegno orario (ad esempio, 20 USD all'ora), quindi aggiungilo al carrello. Gli sconti sono applicati automaticamente all'utilizzo idoneo. Acquista un importo ridotto di sconti a fronte di impegni a cicli regolari, ad esempio ogni 2 settimane o ogni mese. Implementa istanze Spot per carichi di lavoro che possono essere interrotti o che sono stateless. Infine, seleziona le istanze Amazon EC2 on demand e alloca le risorse per i requisiti rimanenti.
- Ciclo di revisione del carico di lavoro: implementa un ciclo di revisione per il carico di lavoro che analizzi in modo specifico la copertura del modello di prezzo. Quando il carico di lavoro ha la copertura necessaria, acquista ulteriori sconti a fronte di impegni parzialmente (ogni tanto) o al variare dell'utilizzo dell'organizzazione.

Risorse

Documenti correlati:

- Introduzione ai consigli Savings Plans
- Accessing Reserved Instance recommendations
- Come acquistare istanze riservate

- Opzioni di acquisto delle istanze
- Istanze Spot
- Modelli di prenotazione per altri servizi AWS
- Servizi supportati dai Savings Plans

Video correlati:

- Risparmia fino al 90% ed esegui i carichi di lavoro di produzione su Spot

Esempi correlati:

- Cosa devo considerare prima di acquistare Savings Plans?
- Come faccio a utilizzare Cost Explorer per analizzare le mie spese e il mio utilizzo?

COST07-BP05 Esecuzione dell'analisi del modello di prezzo a livello di account di gestione

Verifica gli strumenti di gestione dei costi e di fatturazione e dai un'occhiata agli sconti suggeriti con impegni e prenotazioni per eseguire analisi regolari a livello di account di gestione.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

L'esecuzione della modellazione dei costi a intervalli regolari garantisce l'implementazione di opportunità di ottimizzazione su più carichi di lavoro. Ad esempio, se più carichi di lavoro utilizzano istanze on demand a livello aggregato, il rischio di modifica è inferiore e l'implementazione di uno sconto a fronte di impegni permetterà di raggiungere un costo complessivo inferiore. Si consiglia di eseguire l'analisi seguendo cicli regolari con una periodicità da due settimane a un mese. In questo modo è possibile effettuare acquisti in piccoli incrementi, così che la copertura dei modelli di prezzo evolva di pari passo con i carichi di lavoro e i relativi componenti.

Utilizza lo strumento per i suggerimenti AWS Cost Explorer per trovare opportunità di sconti a fronte di impegni nell'account di gestione. I suggerimenti a livello di account di gestione sono calcolati considerando l'utilizzo di tutti gli account nella tua organizzazione AWS con Istanze riservate o la condivisione di sconti Savings Plans (SP) abilitata. Vengono inoltre calcolati quando viene attivata la condivisione degli sconti per consigliare un impegno che massimizzi i risparmi su tutti gli account.

Sebbene in molti casi l'acquisto a livello di account di gestione rappresenti un'ottimizzazione che garantisce risparmi massimi, in alcuni casi potresti prendere in considerazione l'acquisto di Savings Plans a livello di account collegato, ad esempio quando desideri che gli sconti si applichino prima all'utilizzo in quel particolare account collegato. I suggerimenti degli account membri sono calcolati a livello di singolo account per massimizzare i risparmi per ogni account isolato. Se il tuo account ha vincoli o impegni sia per istanze riservate (RI) che per Savings Plans (SP), questi verranno applicati nel seguente ordine:

1. RI zonale
2. RI standard
3. RI convertibile
4. Piano di risparmio delle istanze
5. Piano di risparmio di calcolo

Se acquisti un SP a livello di account di gestione, i risparmi verranno applicati in base alla percentuale di sconto dalla più alta alla più bassa. I Savings Plans a livello di account di gestione esaminano tutti gli account collegati e applicano i risparmi ovunque lo sconto sia il più elevato. Se desideri limitare il luogo in cui vengono applicati i risparmi, puoi acquistare un Savings Plan a livello di account collegato e ogni volta che l'account esegue servizi di calcolo idonei, verrà applicato lo sconto. Quando l'account non esegue servizi di calcolo idonei, lo sconto verrà condiviso con gli altri account collegati con lo stesso account di gestione. La condivisione degli sconti è attivata per impostazione predefinita, ma può essere disattivata se necessario.

In una famiglia con fatturazione consolidata, i Savings Plans vengono applicati prima all'utilizzo dell'account del proprietario e, quindi, all'utilizzo degli altri account. Ciò si verifica solo se la condivisione è abilitata. I tuoi Savings Plans vengono applicati prima alla percentuale di risparmio più alta. Se ci sono più utilizzi con percentuali di risparmio uguali, i Savings Plans sono applicati al primo utilizzo con la tariffa Savings Plans più bassa. I Savings Plans continuano ad essere applicati fino a quando non ci sono più utilizzi rimanenti o fino all'esaurimento dell'impegno o del vincolo. L'eventuale utilizzo residuo viene addebitato in base alle tariffe on demand. Puoi aggiornare i suggerimenti dei Savings Plans in AWS Cost Managemet per generare nuovi suggerimenti dei Savings Plans in qualsiasi momento.

Dopo aver analizzato la flessibilità delle istanze, puoi prendere una decisione in base ai suggerimenti ricevuti. Crea una modellazione dei costi analizzando i costi a breve termine del carico di lavoro rispetto a potenziali diverse opzioni di risorse, analizzando i modelli di prezzo AWS e allineandoli ai requisiti aziendali per scoprire il costo totale di proprietà e le possibilità di Ottimizzazione dei costi .

Passaggi dell'implementazione

Esecuzione di un'analisi degli sconti a fronte di impegni: utilizza Cost Explorer nel tuo account, esamina Savings Plans e le raccomandazioni relative alle istanze riservate. Verifica di aver compreso i suggerimenti dei Saving Plan, fai una stima della tua spesa mensile e calcola il risparmio che puoi ottenere su tale intervallo di tempo. Esamina i consigli a livello di account di gestione, calcolati considerando l'utilizzo in tutti gli account membri della tua organizzazione AWS con abilitata la condivisione degli sconti Savings Plans o Istanze Riservate, per ottenere il massimo risparmio tra gli account. Per assicurarti di implementare le raccomandazioni corrette con gli sconti e i rischi richiesti, segui i Well-Architected Labs.

Risorse

Documenti correlati:

- Come funzionano i prezzi di AWS?
- Opzioni di acquisto dell'istanza
- Panoramica del Saving Plan
- Suggerimenti per il Saving Plan
- Accesso alle raccomandazioni di istanza riservata
- Understanding your Saving Plans recommendation
- Come Savings Plans si applica al tuo utilizzo di AWS
- Saving Plans with Consolidated Billing
- Attivazione delle istanze riservate condivise e degli sconti dei Savings Plans

Video correlati:

- Risparmia fino al 90% ed esegui i carichi di lavoro di produzione su Spot

Esempi correlati:

- AWS Well-Architected Lab: Pricing Models (Level 200)
- AWS Well-Architected Labs: Pricing Model Analysis (Level 200)
- Cosa devo considerare prima di acquistare un Savings Plan?
- How can I use rolling Savings Plans to reduce commitment risk?
- Quando utilizzare le istanze spot

COST 8. In che modo pianifichi i costi per il trasferimento dei dati?

Assicurati di pianificare e monitorare i costi di trasferimento dei dati in modo da poter prendere decisioni sull'architettura per ridurre al minimo i costi. Una modifica piccola ma efficace dell'architettura può ridurre drasticamente i costi operativi nel tempo.

Best practice

- COST08-BP01 Esecuzione della modellazione del trasferimento dei dati
- COST08-BP02 Selezione dei componenti per ottimizzare il costo di trasferimento dei dati
- COST08-BP03 Implementazione dei servizi per ridurre il costo di trasferimento dei dati

COST08-BP01 Esecuzione della modellazione del trasferimento dei dati

Raccogli i requisiti dell'organizzazione ed esegui la modellizzazione del trasferimento dei dati del carico di lavoro e di ciascuno dei suoi componenti. Questo identifica il punto di costo più basso per le sue attuali esigenze di trasferimento dei dati.

Livello di rischio associato alla mancata adozione di questa best practice: elevato

Guida all'implementazione

Quando si progetta una soluzione nel cloud, i costi del trasferimento dei dati vengono in genere ignorati a causa dell'abitudine di progettare l'architettura utilizzando data center on-premise o per mancanza di conoscenze. I costi del trasferimento dei dati in AWS sono determinati dall'origine, dalla destinazione e dal volume del traffico. Tenere conto di questi costi durante la fase di progettazione può produrre risparmi. Capire dove avviene il trasferimento dei dati nel carico di lavoro, il costo del trasferimento e i vantaggi associati è molto importante per stimare con precisione il costo totale di proprietà (TCO). In questo modo puoi prendere una decisione consapevole quando si tratta di modificare o accettare una decisione relativa all'architettura. Ad esempio, potresti disporre di una configurazione con più zone di disponibilità dove replichi i dati tra le varie zone di disponibilità.

Puoi modellare i componenti dei servizi che trasferiscono i dati nel carico di lavoro e decidere che si tratta di un costo accettabile (simile a quello del calcolo e dell'archiviazione in entrambe le zone di disponibilità) per ottenere l'affidabilità e la resilienza richieste. Modella i costi in base a livelli differenti di utilizzo. L'utilizzo del carico di lavoro può cambiare nel corso del tempo e servizi differenti possono risultare più convenienti a livelli differenti.

Mentre modelli il trasferimento dei dati, pensa alla quantità di dati acquisiti e alla loro provenienza. Inoltre, considera la quantità di dati elaborati e la capacità di archiviazione o calcolo necessaria.

Durante la modellazione, attieniti alle best practice relative alle reti in relazione all'architettura del carico di lavoro per ottimizzare i potenziali costi di trasferimento dei dati.

AWS Pricing Calculator può aiutarti a vedere i costi stimati per servizi AWS specifici e per il trasferimento di dati previsto. Se hai un carico di lavoro già in esecuzione (a scopo di test o in un ambiente di preproduzione), usa AWS Cost Explorer o AWS Cost and Usage Report (CUR) per comprendere e modellare i costi di trasferimento dei dati. Configura un proof of concept (PoC) o testa il carico di lavoro ed esegui un test con un carico simulato realistico. Puoi modellare i costi in base alle diverse esigenze di carico di lavoro.

Passaggi dell'implementazione

- Identificazione dei requisiti: quali sono l'obiettivo principale e i requisiti aziendali per il trasferimento pianificato dei dati tra origine e destinazione? Qual è il risultato aziendale previsto finale? Acquisisci i requisiti aziendali e definisci il risultato previsto.
- Identificazione dell'origine e della destinazione: quali sono l'origine e la destinazione dei dati da trasferire? (ad esempio all'interno delle Regioni AWS, verso servizi AWS o in Internet)
  - Trasferimento dei dati all'interno di una Regione AWS
  - Trasferimento dei dati tra Regioni AWS
  - Trasferimento dei dati su Internet
- Identificazione delle classificazioni dei dati: qual è la classificazione dei dati per questo trasferimento di dati? Di che tipo di dati si tratta? Quali sono le dimensioni dei dati? Con quale frequenza devono essere trasferiti i dati? I dati sono sensibili?
- Identificazione dei servizi o degli strumenti AWS da utilizzare: quali servizi AWS vengono utilizzati per questo trasferimento di dati? È possibile utilizzare un servizio che è già stato predisposto per un altro carico di lavoro?
- Calcolo dei costi del trasferimento dei dati: utilizza i prezzi di AWS per la modellazione del trasferimento dei dati creata in precedenza per calcolare i costi di trasferimento dei dati per il carico di lavoro. Calcola i costi di trasferimento dei dati a diversi livelli di utilizzo, ipotizzando incrementi e riduzioni dell'utilizzo del carico di lavoro. Nei casi in cui sono disponibili più opzioni per l'architettura del carico di lavoro valuta i costi di ogni opzione per il confronto.
- Collegamento dei costi ai risultati: per il costo sostenuto per ogni trasferimento dei dati, specifica il risultato ottenuto per il carico di lavoro. Se si tratta di un trasferimento tra componenti potrebbe trattarsi di una necessità di disaccoppiamento, se si tratta di un trasferimento tra zone di disponibilità potrebbe trattarsi di una necessità di ridondanza.

- Creazione della modellazione per il trasferimento dei dati: dopo aver acquisito tutte le informazioni, crea una base concettuale di modellazione del trasferimento dei dati per più casi d'uso e diversi carichi di lavoro.

Risorse

Documenti correlati:

- Soluzioni di memorizzazione nella cache di AWS
- Prezzi di AWS
- Prezzi di Amazon EC2
- Prezzi di Amazon VPC
- Introduzione ai costi di trasferimento dei dati

Video correlati:

- Monitoring and Optimizing Your Data Transfer Costs
- Introduction to Amazon S3 Transfer Acceleration

Esempi correlati:

- Panoramica dei costi di trasferimento dei dati per architetture comuni
- Guida prescrittiva di AWS per le reti

COST08-BP02 Selezione dei componenti per ottimizzare il costo di trasferimento dei dati

Tutti i componenti sono selezionati e l'architettura è progettata per ridurre i costi di trasferimento dei dati. Questo include l'utilizzo di componenti come l'ottimizzazione WAN e le configurazioni Multi-AZ

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Una progettazione basata sul trasferimento dei dati riduce i costi del trasferimento stesso. Potrebbe implicare l'uso di reti di distribuzione di contenuti per posizionare i dati vicino agli utenti, oppure l'uso di collegamenti di rete dedicati dalle tue sedi ad AWS. Puoi anche utilizzare l'ottimizzazione WAN e l'ottimizzazione delle applicazioni per ridurre la quantità di dati trasferiti tra i componenti.

Quando si trasferiscono dati verso il Cloud AWS o al suo interno, è essenziale conoscere la destinazione in base a vari casi d'uso, alla natura dei dati e alle risorse di rete disponibili al fine di selezionare i servizi AWS corretti per ottimizzare il trasferimento dei dati. AWS offre una gamma di servizi personalizzati per le diverse esigenze di migrazione dei dati. Seleziona le opzioni di archiviazione dei dati e trasferimento dei dati corrette in base alle esigenze aziendali della tua organizzazione.

Quando pianifichi o rivedi l'architettura di un carico di lavoro, considera quanto segue:

- Usa gli endpoint VPC in AWS: gli endpoint VPC consentono connessioni private tra il VPC e i servizi AWS supportati. Ciò consente di evitare l'utilizzo della rete Internet pubblica, che può comportare costi di trasferimento dei dati.
- Usa un gateway NAT: utilizza un gateway NAT in modo che le istanze in una sottorete privata possano connettersi a Internet o ai servizi esterni al tuo VPC. Verifica se le risorse dietro il gateway NAT che inviano la maggior parte del traffico si trovano nella stessa zona di disponibilità del gateway NAT. In caso negativo, crea nuovi gateway NAT nella stessa zona di disponibilità della risorsa per ridurre i costi di trasferimento dei dati tra zone di disponibilità.
- L'uso di AWS Direct Connect AWS Direct Connect ignora la rete Internet pubblica e stabilisce una connessione privata diretta tra la rete locale e AWS. Ciò può essere più conveniente e coerente rispetto al trasferimento di grandi volumi di dati su Internet.
- Evita il trasferimento di dati oltre i confini regionali: i trasferimenti di dati tra Regioni AWS (da una regione all'altra) in genere sono a pagamento. Seguire questo approccio basato sul trasferimento tra regioni dovrebbe essere una decisione molto ponderata. Per maggiori dettagli, consulta Scenari multi-regione.
- Monitora il trasferimento dei dati: utilizza Amazon CloudWatch e i log dei flussi VPC per acquisire dettagli sul trasferimento dei dati e sull'utilizzo della rete. Analizza le informazioni sul traffico di rete acquisite nei tuoi VPC, come l'indirizzo IP o l'intervallo a livello di interfacce di rete.
- Analizza l'utilizzo della rete: utilizza strumenti di misurazione e segnalazione come AWS Cost Explorer, i dashboard CUDOS o CloudWatch, per analizzare il costo del trasferimento dei dati del tuo carico di lavoro.

Passaggi dell'implementazione

- Seleziona i componenti per il trasferimento dei dati: utilizzando la modellazione per il trasferimento dei dati descritta in COST08-BP01 Esecuzione della modellazione del trasferimento dei dati, concentrati su dove si trovano i costi di trasferimento dei dati più elevati o dove sarebbero se

l'utilizzo del carico di lavoro cambiasse. Individua architetture alternative o componenti aggiuntivi che eliminano o riducono la necessità di trasferimento dei dati o ne riducono i costi.

Risorse

Best practice correlate:

- COST08-BP01 Esecuzione della modellazione del trasferimento dei dati
- COST08-BP03 Implementazione dei servizi per ridurre il costo di trasferimento dei dati

Documenti correlati:

- Migrazione cloud dei dati
- Soluzioni di memorizzazione nella cache di AWS
- Distribuisci contenuti più rapidamente con Amazon CloudFront

Esempi correlati:

- Panoramica dei costi di trasferimento dei dati per architetture comuni
- Suggerimenti per l'ottimizzazione della rete AWS
- Ottimizzazione delle prestazioni e riduzione dei costi per l'analisi della rete con log di flusso VPC in formato Apache Parquet

COST08-BP03 Implementazione dei servizi per ridurre il costo di trasferimento dei dati

Implementa i servizi per ridurre il costo di trasferimento dei dati. Ad esempio, utilizza edge location o reti per la distribuzione di contenuti (CDN) per fornire contenuti agli utenti finali, crea livelli di memorizzazione nella cache davanti ai database o ai server delle applicazioni e utilizza connessioni di rete dedicate anziché VPN per la connettività al cloud.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Esistono diversi servizi AWS che possono aiutarti a ottimizzare l'utilizzo del trasferimento dei dati di rete. A seconda dei componenti del carico di lavoro, del tipo e dell'architettura cloud, questi

servizi possono aiutarti nella compressione, nella memorizzazione nella cache, nella condivisione e distribuzione del traffico sul cloud.

- Amazon CloudFront è una rete globale di distribuzione di contenuti che trasferisce i dati con una latenza ridotta e una velocità di trasferimento elevata. Cattura i dati nelle posizioni edge di tutto il mondo, riducendo così il carico sulle tue risorse. Utilizzando CloudFront puoi ridurre l'impegno amministrativo legato alla distribuzione dei contenuti per numeri elevati di utenti a livello globale, con una latenza minima. Al security savings bundle può aiutarti a risparmiare fino al 30% sull'utilizzo di CloudFront se prevedi di aumentarlo nel tempo.
- AWS Direct Connect ti consente di creare una connessione di rete dedicata ad AWS. In questo modo puoi ridurre i costi di rete, aumentare la larghezza di banda e offrire un'esperienza di rete più costante rispetto alle connessioni Internet.
- AWS VPN consente di stabilire una connessione sicura e privata tra la rete privata e la rete globale AWS. È ideale per piccoli uffici o partner aziendali perché offre una connettività semplificata ed è un servizio completamente gestito ed elastico.
- Endpoint VPC consentono la connettività tra i servizi AWS su reti private e possono essere utilizzati per ridurre i costi di trasferimento di dati pubblici e dei costi dei Gateway NAT . Gli endpoint VPC del gateway non prevedono tariffe orarie e supportano Amazon S3 e Amazon DynamoDB. Gli endpoint VPC dell'interfaccia sono forniti da AWS PrivateLink e prevedono una tariffa oraria e un costo di utilizzo per GB.
- gateway NAT offrono scalabilità e gestione integrate che riducono i costi rispetto a un'istanza NAT autonoma. Per ridurre i costi di trasferimento ed elaborazione dei dati, posiziona i gateway NAT nelle stesse zone di disponibilità delle istanze a elevato traffico e valuta la possibilità di utilizzare gli endpoint VPC per le istanze che devono accedere a Amazon DynamoDB o Amazon S3.
- utilizza AWS Snow Family dispositivi che dispongono di risorse di calcolo per raccogliere ed elaborare dati all'edge. Dispositivi AWS Snow Family (Snowcone, Snowball e Snowmobile) consentono di spostare petabyte di dati a Cloud AWS in modo conveniente e offline.

Passaggi dell'implementazione

- Implementa i servizi: Seleziona i servizi di rete di AWS applicabili in base al servizio e al tipo di carico di lavoro utilizzando la modellazione del trasferimento dei dati e la revisione dei log di flusso VPC. Scopri dove si trovano i costi maggiori e i flussi con volumi più elevati. Esamina i servizi AWS e valuta se esiste un servizio che riduce o rimuove il trasferimento, in particolare nell'ambito delle reti e della distribuzione di contenuti. Individua anche servizi di caching in cui si verifica un accesso ripetuto ai dati o in cui sono presenti grandi quantità di dati.

Risorse

Documenti correlati:

- AWS Direct Connect
- Esplora i prodotti AWS
- Soluzioni di memorizzazione nella cache AWS
- Amazon CloudFront
- AWS Snow Family
- Amazon CloudFront Security Savings Bundle

Video correlati:

- Monitoring and Optimizing Your Data Transfer Costs
- AWS Cost Optimization Series: CloudFront
- How can I reduce data transfer charges for my NAT gateway?

Esempi correlati:

- How-to chargeback shared services: An AWS Transit Gateway example
- Understand AWS data transfer details in depth from cost and usage report using Athena query and QuickSight
- Panoramica dei costi di trasferimento dei dati per architetture comuni
- Using AWS Cost Explorer to analyze data transfer costs
- Cost-Optimizing your AWS architectures by utilizing Amazon CloudFront features
- How can I reduce data transfer charges for my NAT gateway?

Gestione delle risorse di domanda e offerta

Domanda

- COST 9. Come gestisci la domanda e fornisci le risorse?

COST 9. Come gestisci la domanda e fornisci le risorse?

Per avere un carico di lavoro con costo e prestazioni bilanciate, verifica che venga utilizzato tutto ciò per cui paghi ed evita le istanze molto sottoutilizzate. Un parametro di utilizzo distorto, in qualsiasi delle suddette direzioni, ha un impatto negativo sull'organizzazione, sia per i costi operativi (basse prestazioni a causa di un utilizzo eccessivo) che per le spese inerenti a AWS sprecate (a causa di un provisioning eccessivo).

Best practice

- COST09-BP01 Analisi della domanda del carico di lavoro
- COST09-BP02 Implementazione di un buffer o del throttling per gestire la domanda
- COST09-BP03 Fornitura dinamica delle risorse

COST09-BP01 Analisi della domanda del carico di lavoro

Analizza la domanda del carico di lavoro nel tempo. Verifica che l'analisi copra l'andamento stagionale e rappresenti accuratamente le condizioni operative per l'intera durata del carico di lavoro. L'attività di analisi deve riflettere i potenziali benefici, ad esempio che il tempo speso sia proporzionale al costo del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: alto

Guida all'implementazione

L'analisi della domanda di carichi di lavoro per il cloud computing implica la comprensione dei modelli e delle caratteristiche delle attività di elaborazione avviate nell'ambiente cloud. Questa analisi aiuta gli utenti a ottimizzare l'allocazione delle risorse, gestire i costi e verificare che le prestazioni soddisfino i livelli richiesti.

Conoscere i requisiti del carico di lavoro. I requisiti dell'organizzazione devono indicare i tempi di risposta del carico di lavoro per le richieste. Il tempo di risposta può essere utilizzato per determinare se la domanda è gestita o se l'offerta di risorse cambierà per soddisfare la domanda.

L'analisi deve includere la prevedibilità e la ripetibilità della domanda, la velocità di variazione della domanda e la quantità di variazione della domanda. Esegui l'analisi per un periodo sufficientemente lungo da incorporare qualsiasi variazione stagionale, ad esempio l'elaborazione di fine mese o i picchi legati alle festività.

Lo sforzo di analisi dovrebbe riflettere i potenziali vantaggi dell'implementazione della scalabilità. Osserva il costo totale previsto del componente ed eventuali aumenti o riduzioni di utilizzo e costi durante il ciclo di vita del carico di lavoro.

Di seguito sono riportati alcuni aspetti chiave da prendere in considerazione quando si esegue l'analisi della domanda del carico di lavoro per il cloud computing:

1. Metriche relative all'utilizzo delle risorse e alle prestazioni: analizza come vengono utilizzate le risorse AWS nel tempo. Determina i modelli di utilizzo di picco e non di picco per ottimizzare l'allocazione delle risorse e le strategie di dimensionamento. Monitora i parametri metriche delle prestazioni come tempi di risposta, latenza, throughput e tassi di errore. Queste metriche aiutano a valutare lo stato e l'efficienza complessive dell'infrastruttura cloud.
2. Comportamento di scalabilità di utenti e applicazioni: comprendi il comportamento degli utenti e come influisce sulla domanda del carico di lavoro. L'esame dei modelli di traffico degli utenti aiuta a migliorare la fornitura di contenuti e la reattività delle applicazioni. Analizza la modalità di dimensionamento dei carichi di lavoro in base all'aumento della domanda. Determina se i parametri di dimensionamento automatico sono configurati correttamente ed efficacemente per gestire le fluttuazioni del carico.
3. Tipi di carico di lavoro: identifica i diversi tipi di carichi di lavoro in esecuzione nel cloud, come l'elaborazione in batch, l'elaborazione dei dati in tempo reale, le applicazioni web, i database o i processi di machine learning. Ogni tipo di carico di lavoro può avere requisiti di risorse e profili di prestazioni diversi.
4. Accordi sul livello di servizio (SLA): confronta le prestazioni effettive con gli SLA per garantire la conformità e identificare le aree che necessitano di miglioramento.

Puoi utilizzare Amazon CloudWatch per raccogliere e tenere traccia dei parametri, monitorare i file di log, impostare avvisi e reagire automaticamente ai cambiamenti nelle tue risorse AWS. Puoi anche utilizzare Amazon CloudWatch per ottenere visibilità a livello di sistema su utilizzo delle risorse, prestazioni delle applicazioni e stato di integrità operativa.

con AWS Trusted Advisor, puoi rendere disponibili le tue risorse seguendo le best practice per migliorare le prestazioni e l'affidabilità del sistema, aumentare la sicurezza e trovare opportunità di risparmio di denaro. Puoi anche disattivare le istanze non di produzione e utilizzare Amazon CloudWatch e Auto Scaling per far fronte agli aumenti o alle riduzioni della domanda.

Infine, puoi usare AWS Cost Explorer oppure Amazon QuickSight con il file AWS Cost and Usage Report CUR o i log delle applicazioni per eseguire un'analisi avanzata della domanda del carico di lavoro.

Nel complesso, un'analisi completa della domanda dei carichi di lavoro consente alle organizzazioni di prendere decisioni informate sul provisioning, la scalabilità e l'ottimizzazione delle risorse, con conseguente miglioramento delle prestazioni, dell'efficienza dei costi e della soddisfazione degli utenti.

Passaggi dell'implementazione

- Analizza i dati del carico di lavoro esistenti: Analizza i dati provenienti dal carico di lavoro esistente, dalle versioni precedenti del carico di lavoro o dai modelli di utilizzo previsti. Utilizza Amazon CloudWatch, i file di log e i dati di monitoraggio per ottenere informazioni dettagliate su come è stato utilizzato il carico di lavoro. Analizza un ciclo completo del carico di lavoro e raccogli i dati per eventuali variazioni stagionali, ad esempio eventi di fine mese o di fine anno. L'attività che emerge dall'analisi deve riflettere le caratteristiche del carico di lavoro. L'impegno maggiore dovrebbe riguardare i carichi di lavoro di alto valore che presentano le maggiori variazioni della domanda. Il minimo impegno dovrebbe riguardare carichi di lavoro di basso valore che hanno variazioni minime nella domanda.
- Esegui previsioni dell'influenza dei fattori esterni: Incontra i membri del team di tutta l'organizzazione che possono influenzare o modificare la domanda del carico di lavoro. I team più comuni sono le vendite, il marketing o il business development. Collabora con loro per conoscere i cicli secondo cui operano e se ci sono eventi che potrebbero modificare la domanda del carico di lavoro. Prevedi la richiesta del carico di lavoro con questi dati.

Risorse

Documenti correlati:

- Amazon CloudWatch
- AWS Trusted Advisor
- AWS X-Ray
- AWS Auto Scaling
- AWS Instance Scheduler
- Nozioni di base su Amazon SQS
- AWS Cost Explorer

- Amazon QuickSight

Video correlati:

Esempi correlati:

- Monitor, Track and Analyze for cost optimization
- Searching and analyzing logs in CloudWatch

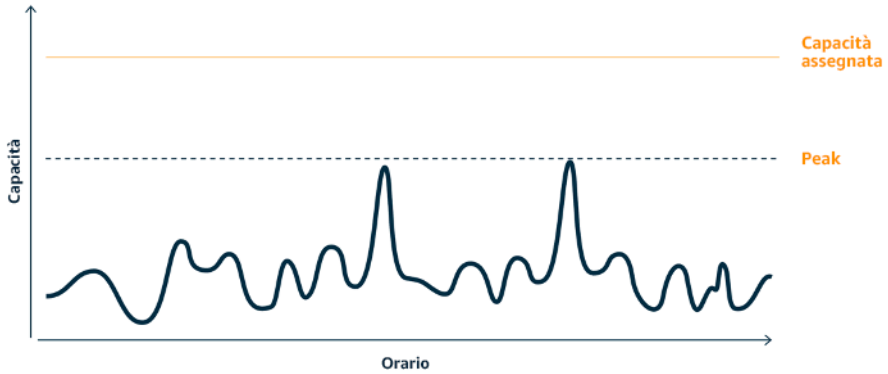
COST09-BP02 Implementazione di un buffer o del throttling per gestire la domanda

Buffering e throttling modificano la domanda sul carico di lavoro, attenuando eventuali picchi. Implementa il throttling quando i client eseguono nuovi tentativi. Implementa il buffering per archiviare la richiesta e rinviare l'elaborazione a un secondo momento. Verifica che le esecuzioni di throttling e buffering siano progettate in modo che i client ricevano una risposta nel tempo richiesto.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

L'implementazione di un buffer o di una limitazione della larghezza di banda della rete è fondamentale nel cloud computing per gestire la domanda e ridurre la capacità allocata richiesta per il carico di lavoro. Per ottenere prestazioni ottimali, è essenziale valutare la domanda totale, compresi i picchi, la velocità con cui variano le richieste e il tempo di risposta necessario. Quando i client hanno la possibilità di inviare nuovamente le proprie richieste, conviene applicare la limitazione della larghezza di banda della rete. Al contrario, per i client che non dispongono della funzionalità di esecuzione di nuovi tentativi, l'approccio ideale è implementare una soluzione buffer. Tali buffer semplificano l'afflusso di richieste e ottimizzano l'interazione delle applicazioni con diverse velocità operative.



Curva della domanda con due picchi distinti che richiedono una capacità elevata

Supponiamo che un carico di lavoro sia caratterizzato dalla curva della domanda illustrata nella figura precedente. Questo carico di lavoro presenta due picchi e per gestire tali picchi viene eseguito il provisioning della capacità di risorse mostrata dalla linea arancione. Le risorse e l'energia utilizzate per questo carico di lavoro non sono indicate nell'area sotto la curva della domanda, ma nell'area sotto la linea della capacità allocata, poiché per gestire questi due picchi è necessario eseguire il provisioning di tale capacità. Diminuire la curva della domanda del carico di lavoro può aiutarti a ridurre la capacità fornita tramite provisioning di un carico di lavoro, oltre al suo impatto sull'ambiente. Per attenuare il picco, valuta la possibilità di implementare una soluzione basata sulla limitazione della larghezza di banda della rete o sul buffering.

Per comprendere meglio queste due soluzioni, proviamo ad analizzarle.

Limitazione della larghezza di banda della rete: se l'origine della richiesta dispone di funzionalità di ripetizione dei tentativi, è possibile implementare la limitazione della larghezza di banda della rete. La limitazione della larghezza di banda della rete indica all'origine che, se non è in grado di soddisfare la richiesta all'ora corrente, dovrebbe riprovare più tardi. L'origine attende un periodo di tempo, quindi riprova a eseguire la richiesta. L'implementazione del throttling ha il vantaggio di limitare la quantità massima di risorse e i costi del carico di lavoro. In AWS, puoi usare Amazon API Gateway per implementare la limitazione della larghezza di banda della rete.

Basato sul buffering: un approccio basato sul buffering utilizza produttori (componenti che inviano messaggi alla coda), consumatori (componenti che ricevono messaggi dalla coda) e una coda (che contiene i messaggi) per archiviare i messaggi. I messaggi vengono letti ed elaborati dai consumatori e ciò consente ai messaggi di essere eseguiti alla velocità che soddisfa i requisiti aziendali del consumatore stesso. Utilizzando una metodologia basata sul buffering, i messaggi dei produttori sono ospitati in code o flussi, dove i produttori possono accedervi a un ritmo in linea con le rispettive esigenze operative.

In AWS, puoi scegliere tra più servizi per implementare un approccio basato sul buffering. Amazon Simple Queue Service(Amazon SQS) è un servizio gestito che fornisce code che consentono a un singolo utente di leggere singoli messaggi. Amazon Kinesis fornisce un flusso che consente a molti utenti di leggere gli stessi messaggi.

Il buffering e la limitazione della larghezza di banda della rete possono attenuare eventuali picchi modificando la domanda sul carico di lavoro. Usa la limitazione della larghezza di banda della rete quando i client riprovano le azioni e usa il buffering per bloccare la richiesta ed elaborarla in un secondo momento. Durante l'utilizzo dell'approccio basato sul buffering, assicurati di progettare il

carico di lavoro per soddisfare la richiesta nel tempo richiesto e verifica di essere in grado di gestire le richieste duplicate. Analizza la domanda complessiva, la velocità di modifica e il tempo di risposta richiesto per determinare le dimensioni del throttling o del buffer richiesto.

Passaggi dell'implementazione

- Analisi dei requisiti del client: analizza le richieste del client per determinare se sono in grado di eseguire nuovi tentativi. Per i client che non possono eseguire nuovi tentativi, è necessario implementare i buffer. Analizza la domanda complessiva, la velocità di modifica e il tempo di risposta richiesto per determinare le dimensioni del throttling o del buffer richiesto.
- Implementazione di un buffer o della limitazione della larghezza di banda della rete: implementa un buffer o la limitazione della larghezza di banda della rete nel carico di lavoro. Una coda come Amazon Simple Queue Service (Amazon SQS) può offrire un buffer ai componenti del carico di lavoro. Amazon API Gateway può fornire una funzionalità di throttling ai componenti del carico di lavoro.

Risorse

Best practice correlate:

- SUS02-BP06 Implementazione del buffering o della limitazione (della larghezza di banda della rete) per ridurre la curva della domanda
- REL05-BP02 Richieste di limitazione (della larghezza di banda della rete)

Documenti correlati:

- AWS Auto Scaling
- AWS Instance Scheduler
- Amazon API Gateway
- Amazon Simple Queue Service
- Nozioni di base su Amazon SQS
- Amazon Kinesis

Video correlati:

- Scegliere il servizio di messaggistica corretto per l'app distribuita

Esempi correlati:

- Gestione e monitoraggio della limitazione delle API nei carichi di lavoro
- Throttling a tiered, multi-tenant REST API at scale using API Gateway
- Enabling Tiering and Throttling in a Multi-Tenant Amazon EKS SaaS Solution Using Amazon API Gateway
- Integrazione dell'applicazione con code e messaggi

COST09-BP03 Fornitura dinamica delle risorse

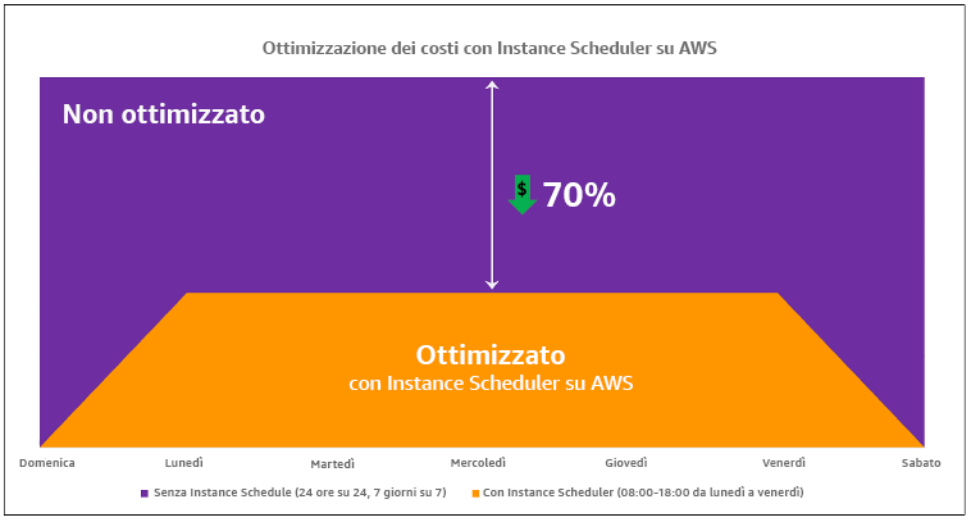
Viene eseguito il provisioning delle risorse in modo pianificato. La pianificazione può essere basata sulla domanda, ad esempio tramite il dimensionamento automatico, oppure sul tempo, quando la domanda è prevedibile e le risorse sono fornite in base al tempo. Questi metodi comportano la minore quantità possibile di provisioning in eccesso o in difetto.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Esistono diversi modi in cui i clienti AWS possono aumentare le risorse disponibili per le proprie applicazioni e fornire risorse per soddisfare la domanda. Una di queste opzioni consiste nell'utilizzare AWS Instance Scheduler, che automatizza l'avvio e l'arresto delle istanze Amazon Elastic Compute Cloud (Amazon EC2) e Amazon Relational Database Service (Amazon RDS). L'altra opzione è utilizzare AWS Auto Scaling, che consente di dimensionare automaticamente le risorse di calcolo in base alla richiesta dell'applicazione o del servizio. Fornire risorse in base alla domanda ti consentirà di pagare solo per le risorse che usi, di ridurre i costi lanciando le risorse quando sono necessarie e di interromperle quando non servono più.

AWS Instance Scheduler consente di configurare l'arresto e l'avvio delle istanze Amazon EC2 e Amazon RDS a orari definiti, in modo da poter soddisfare la domanda delle stesse risorse secondo uno schema orario coerente, ad esempio ogni giorno gli utenti accedono alle istanze Amazon EC2 alle otto del mattino che non servono dopo le sei di sera. Questa soluzione aiuta a ridurre i costi operativi arrestando le risorse inutilizzate e avviandole quando sono necessarie.



Ottimizzazione dei costi con AWS Instance Scheduler.

Puoi anche configurare in modo semplice e rapido le pianificazioni per le tue istanze Amazon EC2 nei tuoi account e nelle tue Regioni con un'interfaccia utente (UI) utilizzando Configurazione rapida di AWS Systems Manager. Puoi pianificare le istanze Amazon EC2 e Amazon RDS con AWS Instance Scheduler e arrestare e avviare le istanze esistenti. Tuttavia, non è possibile arrestare e avviare le istanze che fanno parte del proprio gruppo (ASG) Auto Scaling o che gestiscono servizi come Amazon Redshift o Amazon OpenSearch Service. I gruppi Auto Scaling hanno una propria pianificazione per le istanze del gruppo e queste istanze vengono create.

AWS Auto Scaling ti aiuta a regolare la capacità per mantenere prestazioni stabili e prevedibili al minor costo possibile per soddisfare le mutevoli esigenze. È un servizio gratuito e completamente gestito per il dimensionamento della capacità della tua applicazione, che si integra con le istanze Amazon EC2 e le serie di istanze spot, Amazon ECS, Amazon DynamoDB e Amazon Aurora. Auto Scaling fornisce il rilevamento automatico delle risorse per identificare risorse configurabili nel carico di lavoro, dispone di strategie di dimensionamento integrate volte a ottimizzare le prestazioni, i costi, o trovare un equilibrio tra i due, e fornisce il dimensionamento predittivo per risolvere i picchi ricorrenti con regolarità.

Sono disponibili diverse opzioni di dimensionamento per dimensionare il tuo gruppo Auto Scaling:

- Conservazione dei livelli correnti di istanze in ogni momento
- Dimensionamento manuale
- Dimensionamento basato su una pianificazione
- Dimensionamento basato sulla domanda

- Utilizzo del dimensionamento predittivo

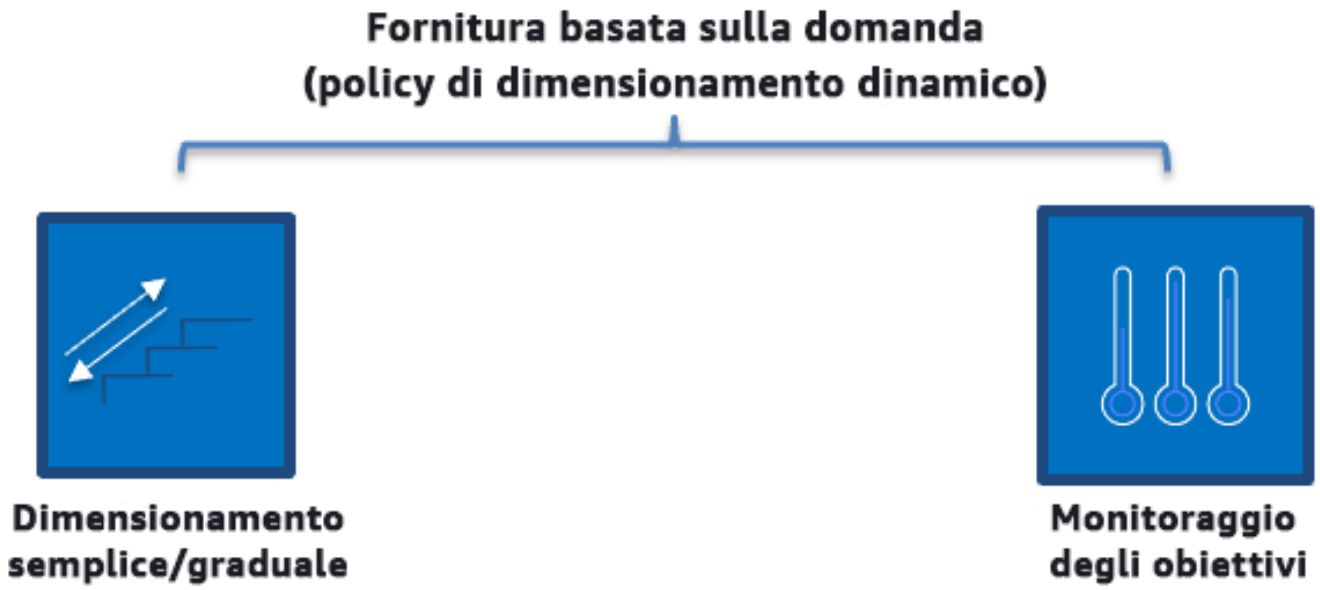
Le policy Auto Scaling sono diverse e possono essere classificate come policy di dimensionamento dinamico e pianificato. Le policy dinamiche fanno riferimento al dimensionamento manuale o dinamico, programmato o predittivo. È possibile utilizzare le policy di dimensionamento per il dimensionamento dinamico, pianificato e predittivo. Puoi anche utilizzare le metriche e gli allarmi di Amazon CloudWatch per attivare eventi di dimensionamento per il tuo carico di lavoro. Ti consigliamo di utilizzare i modelli di avvio, che consentono di accedere alle funzionalità e ai miglioramenti più recenti. Non tutte le funzionalità Auto Scaling sono disponibili quando si utilizzano le configurazioni di avvio. Ad esempio, non è possibile creare un gruppo Auto Scaling che avvii istanze spot e on demand o che specifichi più tipi di istanze. È necessario utilizzare un modello di avvio per configurare queste funzionalità. Quando utilizzi i modelli di avvio, ti consigliamo di modificare ciascuno di essi. Con il controllo delle versioni dei modelli di avvio, puoi creare un sottoinsieme del set completo di parametri. Quindi, puoi riutilizzarlo per creare altre versioni dello stesso modello di avvio.

Puoi utilizzare AWS Auto Scaling o incorporare il dimensionamento nel codice con API o SDK AWS. Ciò riduce i costi complessivi del carico di lavoro rimuovendo i costi operativi dall'apportare manualmente modifiche al tuo ambiente; le modifiche possono essere apportate molto più rapidamente. In questo modo, inoltre, il carico di lavoro viene adattato alla domanda in qualsiasi momento. Per seguire questa best practice e fornire risorse in modo dinamico all'organizzazione, è necessario comprendere il dimensionamento orizzontale e verticale in Cloud AWS e la natura delle applicazioni in esecuzione sulle istanze Amazon EC2. È meglio che il team di Cloud Financial Management collabori con i team tecnici per seguire questa best practice.

Elastic Load Balancing (Elastic Load Balancing) consente di ricalibrare le risorse distribuendo la domanda su più risorse. Utilizzando ASG e Elastic Load Balancing, puoi gestire le richieste in arrivo ottimizzando l'instradamento del traffico in modo che nessuna istanza venga sovraccaricata in un gruppo Auto Scaling. Le richieste vengono distribuite tra tutti gli obiettivi di un gruppo target in modalità Round Robin, senza tenere conto della capacità o dell'utilizzo.

Le metriche tipiche possono essere metriche standard di Amazon EC2, ad esempio l'utilizzo della CPU, la velocità di trasmissione effettiva della rete e la latenza di richiesta/risposta osservata da Elastic Load Balancing. Quando possibile, è consigliabile utilizzare un parametro indicativo dell'esperienza del cliente, in genere si tratta di un parametro personalizzato che potrebbe avere origine dal codice dell'applicazione all'interno del carico di lavoro. Per capire come soddisfare la domanda in modo dinamico in questo documento, Auto Scaling verrà suddiviso in due categorie (modello di fornitura basata sulla domanda e modello di fornitura basata sul tempo) e verrà approfondito ciascun modello.

Fornitura basata sulla domanda: sfrutta l'elasticità del cloud per fornire risorse in grado di soddisfare la domanda in continua evoluzione facendo riferimento allo stato della domanda quasi in tempo reale. Per la fornitura basata sulla domanda, utilizza API o funzionalità dei servizi per modificare in modo programmatico la quantità di risorse del cloud nella tua architettura. Ciò ti consente di dimensionare i componenti nella tua architettura e aumentare il numero di risorse durante i picchi di domanda per mantenere le prestazioni, nonché diminuire la capacità quando la domanda cala in modo da ridurre i costi.

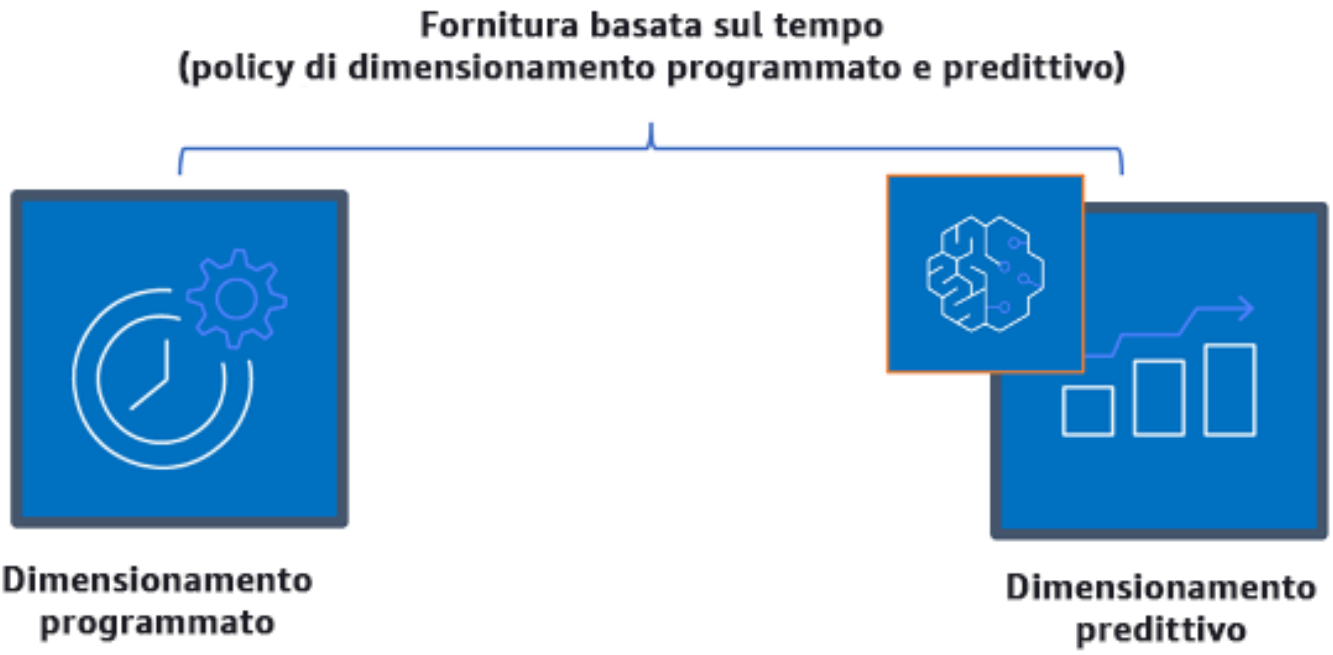


Policy di dimensionamento dinamico basato sulla domanda

- Dimensionamento semplice/graduale: monitora le metriche e aggiunge/rimuove le istanze secondo i passaggi definiti manualmente dai clienti.
- Monitoraggio degli obiettivi: meccanismo di controllo simile a un termostato che aggiunge o rimuove automaticamente le istanze per mantenere le metriche in base a un obiettivo definito dal cliente.

Quando prevedi di usare una strategia basata sulla domanda in un progetto, tieni presenti due considerazioni principali. In primo luogo, devi capire con quale velocità è necessario predisporre le nuove risorse. In secondo luogo, devi capire che la dimensione del margine tra domanda e risorse fornite cambierà. Devi prepararti ad affrontare le variazioni nella domanda, nonché le risorse insufficienti.

Fornitura basata sul tempo: una strategia basata sul tempo allinea la capacità delle risorse alla domanda, che è prevedibile o ben definita nel tempo. In genere questa strategia non dipende dai livelli di utilizzo delle risorse. Una strategia basata sul tempo assicura che le risorse siano disponibili nel momento esatto in cui vengono richieste e possano essere fornite senza ritardi dovuti alle procedure di avvio e ai controlli di sistema o di coerenza. Attraverso una strategia basata sul tempo si possono fornire risorse aggiuntive o incrementare la capacità nei periodi più intensi.



Policy di dimensionamento basato sul tempo

Puoi utilizzare il dimensionamento automatico pianificato e predittivo per implementare un approccio basato sul tempo. I carichi di lavoro possono essere programmati per eseguire il dimensionamento in determinati momenti (ad esempio, all'inizio dell'orario di lavoro), garantendo quindi la disponibilità delle risorse all'arrivo degli utenti on demand. Il dimensionamento predittivo utilizza modelli per dimensionare orizzontalmente, mentre il dimensionamento pianificato utilizza tempi predefiniti per dimensionare orizzontalmente. Puoi anche utilizzare la strategia di selezione del tipo di istanza basata sugli attributi (ABS) nei gruppi Auto Scaling che consenta di esprimere i requisiti dell'istanza come un set di attributi, ad esempio vCPU, memoria e spazio di archiviazione. È possibile utilizzare automaticamente i tipi di istanza di nuova generazione quando vengono rilasciati e accedere a una gamma più ampia di capacità con le istanze spot di Amazon EC2. Il parco istanze Amazon EC2 e Amazon EC2 Auto Scaling selezionano e avviano istanze che corrispondono agli attributi specificati, eliminando la necessità di scegliere manualmente i tipi di istanza.

Puoi anche utilizzare API e SDK AWS e AWS CloudFormation per predisporre e ritirare automaticamente interi ambienti quando ne hai bisogno. Questa strategia risulta particolarmente adatta per gli ambienti di sviluppo o di prova che operano solo in determinati orari di lavoro o periodi di tempo. Puoi usare le API per dimensionare le risorse all'interno di un ambiente (dimensionamento verticale). Ad esempio, potresti dimensionare verticalmente un carico di lavoro di produzione modificando la dimensione o la classe dell'istanza. Ciò è possibile arrestando e avviando l'istanza e selezionando una dimensione o classe diversa. Questa tecnica può essere applicata anche ad altre risorse, come gli Elastic Volumes Amazon EBS, che possono essere modificati per aumentarne le dimensioni, regolarne le prestazioni (IOPS) o modificare il tipo di volume durante l'utilizzo.

Quando prevedi una strategia basata sul tempo in un progetto, tieni presenti due considerazioni principali. In primo luogo, che livello di coerenza presenta il modello di utilizzo? In secondo luogo, qual è l'impatto se il modello cambia? Puoi migliorare l'accuratezza delle previsioni monitorando i tuoi carichi di lavoro e utilizzando la business intelligence. Se noti cambiamenti significativi nel modello di utilizzo, puoi modificare i tempi per assicurarti che la copertura sia fornita.

Passaggi dell'implementazione

- Configura il dimensionamento pianificato: per le variazioni prevedibili della domanda, il dimensionamento basato sul tempo può fornire il numero corretto di risorse in modo tempestivo. Inoltre è utile se la creazione e la configurazione delle risorse non sono abbastanza veloci da rispondere alle variazioni della domanda. Utilizzando l'analisi del carico di lavoro, configura il dimensionamento pianificato utilizzando AWS Auto Scaling. Per configurare la pianificazione basata sul tempo, è possibile utilizzare il dimensionamento predittivo del dimensionamento pianificato per aumentare il numero di istanze Amazon EC2 nei gruppi Auto Scaling in anticipo in base alle variazioni di carico previste o prevedibili.
- Configura il dimensionamento predittivo: il dimensionamento predittivo consente di aumentare il numero di istanze Amazon EC2 del gruppo Auto Scaling in anticipo rispetto agli schemi giornalieri e settimanali dei flussi di traffico. Se si hanno picchi di traffico regolari e applicazioni che richiedono molto tempo per avviarsi, si dovrebbe prendere in considerazione l'utilizzo del dimensionamento predittivo. Il dimensionamento predittivo può aiutare a scalare più velocemente inizializzando la capacità prima del carico previsto rispetto al solo dimensionamento dinamico, che è di natura reattiva. Ad esempio, se gli utenti iniziano a utilizzare il carico di lavoro all'inizio dell'orario di lavoro e non lo utilizzano dopo l'orario di lavoro, il dimensionamento predittivo può aggiungere capacità prima dell'orario di lavoro, eliminando i ritardi del dimensionamento dinamico per reagire alle variazioni del traffico.

- Configura il dimensionamento automatico dinamico: per configurare il dimensionamento in base alle metriche del carico di lavoro attivo, usa Auto Scaling. Utilizza l'analisi e configura Auto Scaling per l'avvio sui livelli di risorse corretti e assicurati che il carico di lavoro si ridimensioni nel tempo richiesto. Si può avviare e dimensionare automaticamente un parco istanze on demand e istanze spot all'interno di un singolo gruppo Auto Scaling. Oltre a ricevere sconti per l'utilizzo delle istanze spot, è possibile utilizzare istanze riservate o Savings Plans per ricevere tariffe scontate rispetto al normale prezzo delle istanze on demand. La combinazione di tutti questi fattori consente di ottimizzare i risparmi sui costi delle istanze Amazon EC2 e di determinare il dimensionamento e le prestazioni desiderate per la tua applicazione.

Risorse

Documenti correlati:

- AWS Auto Scaling
- AWS Instance Scheduler
- Dimensiona il tuo gruppo Auto Scaling
- Nozioni di base su Amazon EC2 Auto Scaling
- Nozioni di base su Amazon SQS
- Dimensionamento programmato per Amazon EC2 Auto Scaling
- Dimensionamento predittivo per Amazon EC2 Auto Scaling

Video correlati:

- Policy di dimensionamento del monitoraggio degli obiettivi per Auto Scaling
- AWS Instance Scheduler

Esempi correlati:

- Selezione del tipo di istanza basata sugli attributi per Auto Scaling per il parco istanze Amazon EC2
- Ottimizzazione dei costi di Amazon Elastic Container Service utilizzando il dimensionamento programmato
- Dimensionamento predittivo con Amazon EC2 Auto Scaling
- Come uso Instance Scheduler con AWS CloudFormation per pianificare le istanze Amazon EC2?

Ottimizzazione nel tempo

Domande

- COST 10. In che modo valuti i nuovi servizi?
- COST 11. Come valuti il costo dell'impegno?

COST 10. In che modo valuti i nuovi servizi?

Nel momento in cui AWS rilascia nuovi servizi e caratteristiche, è buona prassi rivedere le decisioni esistenti relative all'architettura per verificare che continuino a essere le più convenienti.

Best practice

- COST10-BP01 Sviluppo di un processo di revisione del carico di lavoro
- COST10-BP02 Valutazione e analisi regolare del carico di lavoro

COST10-BP01 Sviluppo di un processo di revisione del carico di lavoro

Sviluppa un processo che definisca i criteri e il processo per la revisione del carico di lavoro. L'impegno analitico deve riflettere il potenziale risultato. Ad esempio, i carichi di lavoro principali o i carichi di lavoro con un valore superiore al 10% della fattura sono analizzati trimestralmente oppure ogni sei mesi, mentre i carichi di lavoro inferiori al 10% sono analizzati annualmente.

Livello di rischio associato se questa best practice non fosse adottata: Elevato

Guida all'implementazione

Per far sì che il carico di lavoro sia sempre efficiente in termini di costi, devi analizzarlo regolarmente per stabilire se ci sono opportunità di implementare nuovi servizi, funzionalità e componenti. Per garantire costi complessivi ridotti, il processo deve essere proporzionale al potenziale risparmio. Ad esempio, i carichi di lavoro che rappresentano il 50% della spesa complessiva devono essere esaminati con maggiore regolarità e più nel dettaglio rispetto ai carichi di lavoro che rappresentano il 5% della spesa complessiva. Prendi in considerazione qualsiasi fattore esterno o volatilità. Se il carico di lavoro serve una determinata area geografica o un segmento di mercato e viene previsto un cambiamento in tale area, revisioni più frequenti possono portare a risparmi sui costi. Un altro fattore in fase di revisione è rappresentato dall'impegno necessario per implementare le modifiche. Se i test e la convalida delle modifiche comportassero costi significativi, le revisioni dovrebbero essere meno frequenti.

Prendi in considerazione il costo nel lungo termine della manutenzione di componenti e risorse obsoleti e legacy, e dell'impossibilità di implementare in essi nuove funzionalità. L'attuale costo del test e della convalida potrebbe superare il vantaggio auspicato. Tuttavia, nel corso del tempo, il costo di apportare modifiche potrebbe crescere in modo significativo all'aumentare del divario tra il carico di lavoro e le tecnologie attuali, generando costi ancora maggiori. Ad esempio, il costo del passaggio a un nuovo linguaggio di programmazione potrebbe attualmente non risultare conveniente. Tuttavia, nel giro di cinque anni, il costo del personale qualificato per tale linguaggio potrebbe aumentare e, a causa dell'aumento del carico di lavoro, potresti dover trasferire un sistema ancora più grande al nuovo linguaggio, richiedendo sforzi ancora maggiori rispetto a prima.

Suddividi il carico di lavoro in componenti, assegna un costo ai componenti (una stima è sufficiente) e quindi elenca i fattori (ad esempio, impegno richiesto e mercati esterni) accanto a ciascun componente. Utilizza questi indicatori per determinare una frequenza di revisione per ogni carico di lavoro. Ad esempio, potresti avere i server web come un costo elevato, con un impegno di modifica ridotto e fattori esterni elevati, e da questo potrebbe derivare un'alta frequenza di revisione. Un database centrale può avere un costo medio, con un impegno di modifica elevato e un basso fattore esterno, e da questo potrebbe derivare una frequenza di revisione media.

Definisci un processo per valutare i nuovi servizi, i modelli di progettazione, i tipi di risorse e le configurazioni per ottimizzare il costo del tuo carico di lavoro man mano che diventano disponibili. Simile ai processi di revisione del principio di performance e di revisione del principio di affidabilità, identifica, convalida e assegna la priorità ad attività di ottimizzazione e miglioramento e alla correzione di problematiche e integra questo nel tuo backlog.

Passaggi dell'implementazione

- Definisci la frequenza della revisione: definisci la frequenza con cui il carico di lavoro e i relativi componenti devono essere revisionati. Dedica tempo e risorse al miglioramento continuo e alla frequenza di revisione per migliorare l'efficienza e l'ottimizzazione del carico di lavoro. Si tratta di una combinazione di fattori e può variare da carico di lavoro a carico di lavoro all'interno dell'organizzazione, ma può anche variare tra i componenti del carico di lavoro. Fattori più comuni sono: l'importanza per l'organizzazione misurata in termini di fatturato o marchio, il costo totale di esecuzione del carico di lavoro (inclusi costi operativi e delle risorse), la complessità del carico di lavoro, la facilità di implementazione di una modifica, eventuali accordi di licenza software e l'eventuale aumento dei costi di licenza dovuti a licenze punitive in seguito a una modifica. I componenti possono essere definiti a livello funzionale o tecnico come server Web e database, oppure come risorse di calcolo e storage. Equilibra i fattori di conseguenza e prevedi un periodo per il carico di lavoro e i relativi componenti. Si può decidere di esaminare l'intero carico di lavoro

ogni 18 mesi, esaminare i server Web ogni 6 mesi, il database ogni 12 mesi, l'elaborazione e lo storage a breve termine ogni 6 mesi e lo storage a lungo termine ogni 12 mesi.

- Definisci la completezza della revisione: stabilisci quanto impegno deve essere impiegato per la revisione dei componenti o dell'intero carico di lavoro. Come per la frequenza di revisione, si tratta di un equilibrio tra più fattori. Valuta e dai priorità alle opportunità di miglioramento per concentrare gli sforzi dove producono i vantaggi maggiori, stimando l'impegno necessario per queste attività. Se i risultati non sono in linea con gli obiettivi e l'impegno richiesto ha un costo superiore, riprova utilizzando linee d'azione alternative. I processi di revisione devono prevedere l'allocazione di tempo e risorse per rendere possibile il miglioramento incrementale continuo. Ad esempio, si può decidere di dedicare una settimana all'analisi del componente del database, una settimana di analisi alle risorse di calcolo e quattro ore alla revisione dell'archiviazione.

Risorse

Documenti correlati:

- Blog delle novità di AWS
- Tipi di cloud computing
- Le novità di AWS

Esempi correlati:

- Servizi di supporto proattivo di AWS
- Revisioni costanti dei carichi di lavoro per carichi SAP

COST10-BP02 Valutazione e analisi regolare del carico di lavoro

I carichi di lavoro esistenti vengono rivisti con regolarità in base a ogni processo definito per scoprire se è possibile adottare nuovi servizi, se i servizi esistenti possono essere sostituiti o se i carichi di lavoro possono essere riprogettati.

Livello di rischio associato se questa best practice non fosse adottata: Medium

Guida all'implementazione

AWS aggiunge costantemente nuove funzionalità che ti consentono di innovare e sperimentare in tempi più rapidi con le tecnologie più recenti. Le novità di AWS spiegano nel dettaglio come AWS sta

procedendo in questo senso e offrono una breve panoramica dei servizi AWS, delle funzionalità e degli annunci sulle espansioni a livello regionale nel momento stesso in cui vengono rilasciati. Puoi approfondire i rilasci previsti e usarli per la revisione e l'analisi dei tuoi carichi di lavoro esistenti. Per ottenere i vantaggi offerti dai nuovi servizi e dalle nuove funzionalità di AWS, devi eseguire il processo di revisione sui carichi di lavoro e implementare nuovi servizi e funzionalità in base alle esigenze. Questo significa che potresti aver bisogno di sostituire i servizi esistenti che usi per il tuo carico di lavoro o modernizzare il tuo carico di lavoro per adottare questi nuovi servizi AWS. Ad esempio, puoi esaminare i carichi di lavoro e sostituire il componente di messaggistica con Amazon Simple Email Service. Ciò elimina il costo di gestione e manutenzione di un parco istanze, fornendo al contempo tutte le funzionalità a un costo ridotto.

Per analizzare il tuo carico di lavoro e individuare le opportunità potenziali, dovresti prendere in considerazione non solo i nuovi servizi, ma anche le nuove modalità per creare le soluzioni. Guarda i video della serie Questa è la mia architettura su AWS per scoprire progetti architetturali di altri clienti, le sfide affrontate e le soluzioni adottate. Dai un'occhiata alle serie All-In per scoprire le applicazioni nel mondo reale dei servizi AWS e le storie dei clienti. Puoi anche guardare la serie di video Back to Basics che spiega, esamina e scompone best practice su modelli architetturali cloud di base. Un'altra risorsa è costituita dai video della serie Come si sviluppa, progettati per assistere le persone con grandi idee su come sviluppare un prodotto minimo funzionante (MVP) avvalendosi dei servizi AWS. È un modo per gli sviluppatori di tutto il mondo con grandi idee di ottenere indicazioni sulle architetture da AWS Solutions Architects esperti. Infine, è possibile consultare i materiali della risorsa Nozioni di base con tutorial dettagliati.

Prima di avviare il processo di revisione segui i requisiti aziendali per il carico di lavoro, i requisiti sulla privacy dei dati e la sicurezza per usare un servizio o un'area geografica specifica e i requisiti di performance, seguendo al tempo stesso il processo di revisione concordato.

Passaggi dell'implementazione

- Esamina con regolarità il carico di lavoro: utilizzando il processo definito, esegui le revisioni con la frequenza specificata. Accertati di dedicare la quantità di impegno necessaria per ciascun componente. Questo processo è simile a quello di progettazione iniziale in cui hai selezionato i servizi per l'ottimizzazione dei costi. Analizza i servizi e i vantaggi che porterebbero; questa volta considera anche il costo del tempo necessario per la modifica, non solo i vantaggi a lungo termine.
- Implementa nuovi servizi: se in seguito all'analisi ritieni di dover implementare modifiche, esegui innanzitutto una baseline del carico di lavoro per scoprire il costo corrente per ogni output. Implementa le modifiche, quindi esegui un'analisi per verificare il nuovo costo per ogni output.

Risorse

Documenti correlati:

- Blog delle novità di AWS
- Le novità di AWS
- Documentazione AWS
- Nozioni di base su AWS
- Risorse generali su AWS

Video correlati:

- AWS - La mia architettura
- AWS - Ripartiamo dall'inizio
- AWS - Serie All-In
- Come si sviluppa

COST 11. Come valuti il costo dell'impegno?

Best practice

- COST11-BP01 Esecuzione dell'automazione per le operazioni

COST11-BP01 Esecuzione dell'automazione per le operazioni

Valuta i costi operativi del cloud, concentrandoti sulla quantificazione del risparmio di tempo e impegno nelle attività amministrative e nelle implementazioni, sulla mitigazione del rischio di errore umano, sulla conformità e su altre operazioni tramite l'automazione. Valuta il tempo e i costi associati necessari per gli impegni operativi e implementa l'automazione per le attività amministrative per ridurre al minimo il lavoro manuale laddove possibile.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

L'automazione delle operazioni riduce la frequenza di attività manuali, migliora l'efficienza e offre vantaggi ai clienti con un'esperienza affidabile e coerente durante l'implementazione,

l'amministrazione o l'operatività dei carichi di lavoro. Puoi liberare le risorse dell'infrastruttura dalle attività operative manuali e usarle per operazioni e innovazioni di maggior valore, migliorando così i risultati aziendali. Le aziende vogliono un modo testato e collaudato di gestire i propri carichi di lavoro nel cloud. La soluzione deve essere sicura, veloce e contenuta nei costi, con rischio minimo e massima affidabilità.

Inizia assegnando le priorità alle tue operazioni sulla base dell'impegno richiesto, considerando i costi complessivi. Ad esempio, quanto tempo è necessario per distribuire nuove risorse nel cloud, eseguire modifiche di ottimizzazione alle risorse esistenti o implementare le configurazioni necessarie? Esamina il costo totale delle attività eseguite dal personale, tenendo conto dei costi operativi e di gestione. Dai la priorità alle automazioni per le attività amministrative per ridurre il livello di impegno delle persone.

L'impegno di revisione deve riflettere il potenziale risultato. Ad esempio, esamina il tempo impiegato per eseguire le attività manualmente rispetto a quello per eseguirle in automatico. Dai priorità all'automazione di attività ripetitive e di valore elevato che richiedono tempo e sono complesse. Le attività che presentano un rischio o un valore elevato di errore umano sono in genere il punto di partenza migliore da cui iniziare con l'automazione, poiché il rischio spesso comporta un costo operativo aggiuntivo indesiderato (come gli straordinari del team operativo).

Utilizza gli strumenti di automazione come AWS Systems Manager o AWS Config per semplificare le operazioni, la conformità, il monitoraggio, il ciclo di vita e i processi di terminazione. Con i servizi e gli strumenti AWS nonché i prodotti di terze parti, puoi personalizzare le automazioni che implementi per soddisfare le tue esigenze specifiche. La tabella seguente mostra alcune delle funzioni e delle caratteristiche operative di base che puoi ottenere con i servizi AWS per automatizzare attività amministrative e operative:

- AWS Audit Manager: esegui un audit continuo del tuo utilizzo di AWS per semplificare la valutazione dei rischi e della conformità
- AWS Backup: gestisci centralmente e automatizza la protezione dei dati.
- AWS Config: configura le risorse di elaborazione, valuta, esegui audit e analizza le configurazioni e l'inventario delle risorse.
- AWS CloudFormation: avvia risorse altamente disponibili con infrastructure as code.
- AWS CloudTrail: gestisci le modifiche IT, la conformità e il controllo.
- Amazon EventBridge: pianifica gli eventi e attiva AWS Lambda.
- AWS Lambda: automatizza i processi ripetitivi attivandoli con eventi o eseguendoli sulla base di una pianificazione prefissata con AWS EventBridge.

- AWS Systems Manager: avvia e interrompi carichi di lavoro, applica patch ai sistemi operativi, automatizza la configurazione e la gestione continua.
- AWS Step Functions: pianifica i processi e automatizza i flussi di lavoro.
- AWS Service Catalog: utilizza modelli e infrastructure as code con conformità e controllo.

Se desideri adottare immediatamente le automazioni usando i prodotti e i servizi AWS, ma non hai le risorse richieste al tuo interno, contattaAWS Managed Services (AMS), AWS Professional Services o i partner AWS per promuovere l'adozione dell'automazione e migliorare la tua eccellenza operativa nel cloud.

AWS Managed Services (AMS) è un servizio che gestisce l'infrastruttura AWS per conto di clienti e partner aziendali. Fornisce un ambiente sicuro e conforme in cui è possibile distribuire i carichi di lavoro. AMS utilizza modelli operativi cloud aziendali dotati di automazione per consentirti di soddisfare i requisiti aziendali, di passare più rapidamente al cloud e di ridurre i costi di gestione correnti.

AWS Professional Services può anche aiutarti a raggiungere i risultati aziendali auspicati e ad automatizzare le operazioni con AWS. Consente ai clienti di implementare operazioni IT automatizzate, solide e agili, nonché funzionalità di governance ottimizzate per il cloud. Per esempi di monitoraggio dettagliati e best practice consigliate, consulta il white paper sul principio dell'eccellenza operativa.

Passaggi dell'implementazione

- Sviluppa una volta e implementa molte: usa infrastructure as code come CloudFormation, AWS SDK o AWS CLI per implementare una volta e usare molte volte per lo stesso ambiente o per scenari di ripristino di emergenza. Applica i tag durante l'implementazione per monitorare il tuo consumo definito in altre best practice. Usa AWS Launch Wizard per ridurre i tempi di implementazione di molti carichi di lavoro aziendali diffusi. AWS Launch Wizard ti guida attraverso il dimensionamento, la configurazione e l'implementazione di carichi di lavoro aziendali seguendo le best practice AWS. Puoi anche usare Service Catalog, che ti consente di creare e gestire modelli approvati di infrastructure as code da utilizzare su AWS in modo che tutti abbiano accesso a risorse cloud self-service e approvate.
- Automatizza la conformità continua: puoi automatizzare la valutazione e la correzione delle configurazioni registrate rispetto agli standard predefiniti. Quando combini AWS Organizations con le funzionalità di AWS Config e AWS CloudFormation, puoi gestire e automatizzare in modo efficiente la conformità della configurazione su larga scala per centinaia di account membri. Puoi

esaminare le modifiche delle configurazioni e delle relazioni tra le risorse AWS e approfondire la cronologia di una configurazione di risorsa.

- Automatizza le attività di monitoraggio: AWS fornisce vari strumenti per monitorare i servizi. Puoi configurare questi strumenti per automatizzare le attività di monitoraggio. Crea e implementa un piano di monitoraggio che raccolga i dati da tutte le parti del carico di lavoro in modo da poter eseguire più facilmente il debug di un errore su più punti, se si verifica. Ad esempio, puoi utilizzare gli strumenti di monitoraggio automatizzati per osservare Amazon EC2 e ricevere una segnalazione quando qualcosa non va secondo i controlli dello stato del sistema, i controlli dello stato delle istanze e gli allarmi Amazon CloudWatch.
- Automatizza la manutenzione e le operazioni: esegui operazioni di routine automaticamente senza l'intervento umano. Con i servizi e gli strumenti AWS, puoi scegliere quali automazioni AWS implementare e personalizzare per i tuoi requisiti specifici. Ad esempio, utilizza EC2 Image Builder per la creazione, il test e l'implementazione di immagini di macchine virtuali e container da utilizzare su AWS o on-premises per applicare patch alle istanze EC2 con AWS SSM. Se l'azione desiderata non può essere eseguita con i servizi AWS o se hai bisogno di azioni più complesse con risorse di filtro, allora automatizza le tue operazioni con gli strumenti AWS Command Line Interface (AWS CLI) o AWS SDK. AWS CLI offre la possibilità di automatizzare l'intero processo di controllo e gestione dei servizi AWS tramite script senza usare la AWS Management Console. Seleziona i tuoi AWS SDK preferiti per interagire con i servizi AWS. Per altri esempi di codice, consulta il repository di esempi di AWS SDK Code.
- Crea un ciclo di vita continuo con le automazioni: è importante stabilire e preservare policy consolidate del ciclo di vita non solo per la normativa o la ridondanza, ma anche per l'ottimizzazione dei costi. Puoi utilizzare AWS Backup per gestire e automatizzare centralmente la protezione dei dati degli archivi di dati, come bucket, volumi, database e file system. Puoi inoltre usare Amazon Data Lifecycle Manager per automatizzare la creazione, la conservazione e l'eliminazione degli snapshot EBS e delle AMI basate su EBS.
- Elimina le risorse non necessarie: è abbastanza comune accumulare risorse inutilizzate nella sandbox o negli Account AWS di sviluppo. Gli sviluppatori creano e sperimentano vari servizi e risorse come parte del normale ciclo di sviluppo, quindi non eliminano le risorse quando non sono più necessarie. Le risorse inutilizzate possono comportare costi superflui e talvolta elevati per l'organizzazione. L'eliminazione di queste risorse può ridurre i costi operativi di questi ambienti. Assicurati che i dati non siano necessari o esegui un backup se non sei sicuro. È possibile usare AWS CloudFormation per pulire gli stack implementati, eliminando automaticamente la maggior parte delle risorse definite nel modello. In alternativa, puoi creare un'automazione per l'eliminazione delle risorse AWS utilizzando strumenti come aws-nuke.

Risorse

Documenti correlati:

- Modernizing operations in the Cloud AWS
- AWS Services for Automation
- Infrastructure and automation
- AWS Systems Manager Automation
- Automated and manual monitoring
- AWS automations for SAP administration and operations
- AWS Managed Services
- AWS Professional Services

Video correlati:

- Automate Continuous Compliance at Scale in AWS
- AWS Backup Demo: Cross-Account & Cross-Region Backup
- Patching for your Amazon EC2 Instances

Esempi correlati:

- Reinventing automated operations (Part I)
- Reinventing automated operations (Part II)
- Automate deletion of AWS resources by using aws-nuke
- Delete unused Amazon EBS volumes by using AWS Config and AWS SSM
- Automate continuous compliance at scale in AWS
- IT Automations with AWS Lambda

Sostenibilità

Il principio della sostenibilità include la consapevolezza dell'impatto dei servizi utilizzati, la quantificazione di tale impatto per l'intero ciclo di vita del carico di lavoro e l'applicazione dei principi di progettazione e delle best practice per ridurlo nella fase di sviluppo di carichi di lavoro cloud. È

possibile trovare linee guida prescrittive sull'implementazione nel Whitepaper sul principio della sostenibilità.

Aree delle best practice

- Selezione delle regioni
- Allineamento alla domanda
- Software e architettura
- Dati
- Hardware e servizi
- Processo e cultura

Selezione delle regioni

Domanda

- SUS 1 Come si selezionano le regioni per un carico di lavoro?

SUS 1 Come si selezionano le regioni per un carico di lavoro?

La scelta della Regione per il carico di lavoro influisce in modo significativo sui suoi KPI, tra cui prestazioni, costi e impatto ambientale. Per migliorare in modo efficace questi KPI, devi scegliere le regioni per i carichi di lavoro in base alle esigenze aziendali e agli obiettivi di sostenibilità.

Best practice

- SUS01-BP01 Scelta della Regione in base alle esigenze aziendali e agli obiettivi di sostenibilità.

SUS01-BP01 Scelta della Regione in base alle esigenze aziendali e agli obiettivi di sostenibilità.

Scegli la Regione del tuo carico di lavoro in base alle esigenze aziendali e agli obiettivi di sostenibilità per ottimizzare i suoi KPI, tra cui prestazioni, costi e impatto ambientale.

Anti-pattern comuni:

- Selezione della Regione del carico di lavoro in base alla propria collocazione.
- Consolidamento di tutte le risorse del carico di lavoro in un'unica posizione geografica.

Vantaggi dell'adozione di questa best practice: la collocazione di un carico di lavoro in prossimità di progetti legati alla generazione di energia rinnovabile di Amazon o in Regioni con un'intensità ridotta di emissione di anidride carbonica nota può contribuire a ridurre il suo impatto ambientale.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Il Cloud AWS è una rete in costante espansione di Regioni e punti di presenza (PoP), con un'infrastruttura di rete globale che li collega tra loro. La scelta della Regione per il carico di lavoro influisce in modo significativo sui suoi KPI, tra cui prestazioni, costi e impatto ambientale. Per migliorare efficacemente questi KPI, è necessario scegliere le Regioni per il proprio carico di lavoro in base alle esigenze aziendali e agli obiettivi di sostenibilità.

Passaggi dell'implementazione

- Segui questi passaggi per valutare e selezionare le potenziali Regioni per il tuo carico di lavoro in base ai requisiti aziendali, tra cui la conformità, le funzionalità disponibili, il costo e la latenza.
  - Conferma che queste Regioni siano conformi in base alle normative locali richieste.
  - Utilizza gli elenchi dei servizi regionali di AWS per verificare che le Regioni dispongano dei servizi e delle funzionalità necessarie per gestire il tuo carico di lavoro.
  - Calcola il costo del carico di lavoro su ogni Regione utilizzando AWS Pricing Calculator.
  - Valuta la latenza di rete tra le sedi degli utenti finali e ogni Regione AWS.
- Scegli le Regioni in prossimità dei progetti di generazione di energia rinnovabile di Amazon e le Regioni in cui la griglia presenta un'intensità di emissione di anidride carbonica nota inferiore a quella di altre sedi (o Regioni).
  - Identifica le tue principali linee guida sulla sostenibilità per tracciare e confrontare le emissioni di anidride carbonica anno per anno sulla base del Greenhouse Gas Protocol (metodi basati sul mercato e sulla localizzazione).
  - Scegli la Regione in base al metodo utilizzato per monitorare le emissioni di anidride carbonica. Per maggiori dettagli sulla scelta di una Regione in base alle linee guida sulla sostenibilità, consulta Come selezionare una Regione per il carico di lavoro in base agli obiettivi di sostenibilità.

Risorse

Documenti correlati:

- Comprendere le stime delle emissioni di anidride carbonica
- Amazon Around the Globe
- Metodologia delle energie rinnovabili
- Quali elementi valutare quando si seleziona una Regione per i propri carichi di lavoro

Video correlati:

- AWS re:Invent 2023 - Sustainability innovation in AWS Global Infrastructure
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2022 - Architecting sustainably and reducing your AWS carbon footprint
- AWS re:Invent 2022 - Sustainability in AWS global infrastructure

Allineamento alla domanda

Domanda

- SUS 2 Come si allineano le risorse cloud alla domanda?

SUS 2 Come si allineano le risorse cloud alla domanda?

Il modo in cui gli utenti e le applicazioni utilizzano i carichi di lavoro e altre risorse può aiutarti a identificare i miglioramenti da implementare per realizzare gli obiettivi di sostenibilità. Dimensiona l'infrastruttura in modo che sia costantemente adatta alla domanda e verifica di usare solo le risorse minime necessarie per supportare gli utenti. Allinea i livelli di servizio alle esigenze dei clienti. Colloca le risorse in modo da limitare la rete necessaria per il loro consumo da parte di utenti e applicazioni. Rimuovi gli asset inutilizzati. Offri ai membri del team dispositivi in grado di soddisfarne le esigenze con un impatto minimo in termini di sostenibilità.

Best practice

- SUS02-BP01 Scala dinamicamente l'infrastruttura dei carichi di lavoro
- SUS02-BP02 Allineamento degli SLA agli obiettivi di sostenibilità
- SUS02-BP03 Interruzione della creazione e della manutenzione di risorse inutilizzate
- SUS02-BP04 Ottimizzazione del posizionamento geografico dei carichi di lavoro in base ai requisiti di rete

- SUS02-BP05 Ottimizzazione delle risorse dei membri del team in base alle attività eseguite
- SUS02-BP06 Implementazione del buffering o della limitazione (della larghezza di banda della rete) per ridurre la curva della domanda

SUS02-BP01 Scala dinamicamente l'infrastruttura dei carichi di lavoro

Usa l'elasticità del cloud e dimensiona la tua infrastruttura in modo dinamico per rispondere alla richiesta di fornitura di risorse cloud ed evitare capacità sovra-assegnate nel tuo carico di lavoro.

Anti-pattern comuni:

- Mancato dimensionamento dell'infrastruttura in base al carico degli utenti.
- Costante dimensionamento manuale dell'infrastruttura.
- Dopo un evento di dimensionamento, lasci una capacità aumentata anziché ridurre il dimensionamento.

vantaggi derivanti dall'applicazione di questa best practice: la configurazione e il test dell'elasticità dei carichi di lavoro aiuta ad abbiinare correttamente richiesta e fornitura di risorse cloud e a evitare capacità sovra-assegnate. Puoi sfruttare i vantaggi dell'elasticità nel cloud per dimensionare automaticamente la capacità durante e dopo i picchi di richiesta ed essere sicuro di utilizzare solo il numero esatto di risorse necessario per soddisfare le esigenze aziendali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Il cloud offre la flessibilità necessaria per espandere o ridurre le risorse in modo dinamico attraverso una serie di meccanismi per soddisfare i cambiamenti della domanda. La corrispondenza ottimale tra offerta e domanda consente l'impatto ambientale più basso per un carico di lavoro.

La domanda può essere fissa o variabile e richiede parametri e automazione, allo scopo di garantire che la gestione non diventi particolarmente onerosa. Le applicazioni possono essere dimensionate verticalmente (verso l'alto o verso il basso) modificando la dimensione dell'istanza, orizzontalmente (aumentando o diminuendo) modificando il numero di istanze o tramite una combinazione delle due opzioni.

Puoi adottare varie strategie di approccio per associare l'offerta di risorse alla domanda.

- Approccio di monitoraggio del target: monitora il parametro di dimensionamento e aumenta o diminuisci automaticamente la capacità in base alle esigenze.
- Dimensionamento predittivo: dimensiona l'anticipazione di tendenze giornaliere e settimanali.
- Approccio basato sulla pianificazione: imposta la tua pianificazione di dimensionamento in base a modifiche di carico prevedibili.
- Dimensionamento dei servizi: scegli i servizi (come il serverless) che usano il dimensionamento in modo nativo per impostazione predefinita o che forniscono il dimensionamento automatico come funzionalità.

Identifica i periodi di utilizzo assente o ridotto e dimensiona le risorse per evitare capacità in eccesso e migliorare il livello di efficienza.

Passaggi dell'implementazione

- L'elasticità corrisponde all'offerta di risorse disponibili rispetto alla relativa domanda. Istanze, container e funzioni offrono meccanismi di elasticità, sia insieme al dimensionamento automatico sia come funzionalità del servizio. AWS offre una gamma di meccanismi di dimensionamento automatico per avere la certezza che i carichi di lavoro possano essere ridotti facilmente e velocemente nei periodi di basso carico di utenti. Ecco alcuni esempi di meccanismi di dimensionamento automatico:

Auto scaling mechanism	Where to use
Amazon EC2 Auto Scaling	Usalo per verificare che sia disponibile il numero corretto di istanze Amazon EC2 per gestire il carico degli utenti dell'applicazione.
Application Auto Scaling	Usalo per dimensionare automaticamente le risorse per servizi AWS diversi da Amazon EC2, ad esempio funzioni Lambda o servizi Amazon Elastic Container Service (Amazon ECS).
Kubernetes Cluster Autoscaler	Usalo per dimensionare automaticamente i cluster Kubernetes su AWS.

- Si parla spesso di dimensionamento con servizi di elaborazione come le istanze Amazon EC2 o le funzioni AWS Lambda. Considera la configurazione di servizi non di elaborazione come unità di capacità di lettura e scrittura Amazon DynamoDB o partizioni Amazon Kinesis Data Streams per rispondere alle richieste.
- Verifica che le metriche per il dimensionamento verticale o orizzontale siano convalidate in base al tipo di carico di lavoro implementato. Se distribuisci un'applicazione di transcodifica video, è previsto il 100% di utilizzo della CPU e non deve essere il parametro principale. Se necessario, puoi usare una metrica personalizzata (come l'uso della memoria) per la tua politica di dimensionamento. Per scegliere la metrica corretta, consulta le linee guida seguenti per Amazon EC2:
  - La metrica deve essere una metrica di utilizzo valida e descrivere il livello di impiego di un'istanza.
  - Il valore della metrica deve aumentare o diminuire proporzionalmente in base al numero di istanze nel gruppo con Auto Scaling.
- Usa il dimensionamento dinamico invece del dimensionamento manuale per il tuo gruppo Auto Scaling. Ti consigliamo anche di usare politiche di dimensionamento del monitoraggio degli obiettivi nel tuo dimensionamento dinamico.
- Verifica che le implementazioni dei carichi di lavoro siano in grado di gestire eventi di dimensionamento orizzontale. Crea scenari di test per eventi di dimensionamento orizzontale per verificare che il carico di lavoro si comporti secondo le aspettative e che non incida sull'esperienza utente (come nel caso della perdita di sessioni permanenti). Puoi usare la Cronologia delle attività per verificare un'attività di dimensionamento per un gruppo Auto Scaling.
- Analizza il tuo carico di lavoro per individuare modelli prevedibili e dimensionare le tue risorse in modo proattivo, anticipando variazioni nella domanda previste e pianificate. Con il dimensionamento predittivo puoi eliminare la necessità di offrire capacità in eccedenza. Per maggiori dettagli consulta Dimensionamento predittivo con Amazon EC2 Auto Scaling.

Risorse

Documenti correlati:

- Nozioni di base su Amazon EC2 Auto Scaling
- Dimensionamento predittivo per EC2, alimentato dal machine learning
- Analizza il comportamento degli utenti tramite Amazon OpenSearch Service, Amazon Data Firehose e Kibana

- Che cos'è Amazon CloudWatch?
- Monitoraggio del carico del database con Performance Insights su Amazon RDS
- Introduzione al supporto nativo per il dimensionamento predittivo con Amazon EC2 Auto Scaling
- Introducing Karpenter - An Open-Source, High-Performance Kubernetes Cluster Autoscaler (Introduzione a Karpenter - Kubernetes Cluster Autoscaler, uno strumento open source a elevate prestazioni)
- Deep Dive su Amazon ECS Cluster Auto Scaling

Video correlati:

- AWS re:Invent 2023 - Scaling on AWS for the first 10 million users
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Build a cost-, energy-, and resource-efficient compute environment
- AWS re:Invent 2022 - Scaling containers from one user to millions
- AWS re:Invent 2023 - Scaling FM inference to hundreds of models with Amazon SageMaker
- AWS re:Invent 2023 - Harness the power of Karpenter to scale, optimize & upgrade Kubernetes

Esempi correlati:

- Dimensionamento automatico

SUS02-BP02 Allineamento degli SLA agli obiettivi di sostenibilità

Rivedi e ottimizza gli Accordi sul livello di servizio (SLA) del carico di lavoro in base ai tuoi obiettivi di sostenibilità per ridurre la quantità di risorse richieste per supportare il carico di lavoro e continuare a soddisfare le esigenze aziendali.

Anti-pattern comuni:

- Gli SLA dei carichi di lavoro sono sconosciuti o ambigui.
- Definisci il tuo SLA solo per disponibilità e performance.
- Usi lo stesso modello di progettazione (come l'architettura multi-AZ) per tutti i carichi di lavoro.

Vantaggi dell'adozione di questa best practice: allineare gli SLA agli obiettivi di sostenibilità porta a un utilizzo ottimale delle risorse e, al contempo, a una conciliazione con le esigenze aziendali.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Gli SLA definiscono il livello di servizio che ci si aspetta da un carico di lavoro cloud, come ad esempio i tempi di risposta, la disponibilità e la conservazione dei dati. Essi influenzano l'architettura, l'utilizzo delle risorse e l'impatto ambientale di un carico di lavoro nel cloud. Con cadenza regolare, rivedi gli SLA e accetta dei compromessi che riducano l'utilizzo di risorse in modo significativo in cambio di una diminuzione accettabile dei livelli di servizio.

Passaggi dell'implementazione

- Comprendi gli obiettivi di sostenibilità: individua nella tua organizzazione gli obiettivi di sostenibilità, come la riduzione delle emissioni di carbonio o il miglioramento dell'utilizzo delle risorse.
- Rivedi gli SLA: esamina gli accordi sul livello di servizio (SLA) per valutare se supportano i requisiti aziendali. Se stai superando gli SLA, esegui un'ulteriore revisione.
- Comprendi i compromessi: individua i compromessi tra la complessità del carico di lavoro, ad esempio un elevato volume di utenti simultanei, le prestazioni, ad esempio la latenza, e l'impatto sulla sostenibilità, ad esempio le risorse richieste. In genere, dare la priorità a due fattori va a scapito del terzo.
- Modifica gli SLA: cambia gli SLA accettando compromessi che riducano l'impatto in termini di sostenibilità in modo significativo in cambio di una diminuzione accettabile dei livelli di servizio.
  - Sostenibilità e affidabilità: i carichi di lavoro altamente disponibili tendono a consumare più risorse.
  - Sostenibilità e performance: l'uso di una maggiore quantità di risorse per aumentare le performance potrebbe avere un impatto ambientale più significativo.
  - Sostenibilità e sicurezza: carichi di lavoro con una sicurezza eccessiva potrebbero avere un impatto maggiore sull'ambiente.
- Definisci gli SLA di sostenibilità, se possibile: includi gli SLA di sostenibilità per il tuo carico di lavoro. Ad esempio, determina un livello minimo di utilizzo come SLA di sostenibilità per le tue istanze di calcolo.
- Usa modelli di progettazione efficienti: utilizza modelli di progettazione, come i microservizi su AWS, che danno la priorità a funzioni strategiche per la tua azienda e consentono livelli di servizio inferiori (in tema di obiettivi per tempi di risposta o di ripristino) per funzioni non critiche.

- Comunica e stabilisci le responsabilità: condividi gli SLA con tutte le parti interessate pertinenti, inclusi il team di sviluppo e i tuoi clienti. Utilizza i report per tracciare e monitorare gli SLA. Assegna le responsabilità per raggiungere gli obiettivi di sostenibilità degli SLA.
- Usa incentivi e premi: utilizza incentivi e premi per raggiungere o superare gli SLA in linea con gli obiettivi di sostenibilità.
- Rivedi e itera: revisiona e modifica regolarmente i tuoi SLA per assicurarti che siano allineati agli obiettivi di sostenibilità e prestazioni in costante evoluzione.

Risorse

Documenti correlati:

- Understand resiliency patterns and trade-offs to architect efficiently in the cloud
- L'importanza del contratto sul livello di servizi (SLA) per i provider SaaS

Video correlati:

- AWS re:Invent 2023 - Capacity, availability, cost efficiency: Pick three
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2023 - Advanced integration patterns & trade-offs for loosely coupled systems
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2022 - Build a cost-, energy-, and resource-efficient compute environment

SUS02-BP03 Interruzione della creazione e della manutenzione di risorse inutilizzate

Disattiva le risorse non utilizzate nel tuo carico di lavoro per ridurre il numero di risorse cloud richieste per supportare la domanda e per ridurre gli sprechi.

Anti-pattern comuni:

- Non analizzi la tua applicazione per individuare le risorse ridondanti o non più necessarie.
- Non rimuovi le risorse ridondanti o non più necessarie.

Vantaggi dell'adozione di questa best practice: se si eliminano le risorse non utilizzate si libera capacità e si migliora l'efficienza generale del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Le capacità inutilizzate consumano risorse cloud come spazio di archiviazione e potenza di elaborazione. Individuando ed eliminando queste risorse, puoi liberare capacità e ottenere un'architettura cloud più efficiente. Analizza le risorse delle applicazioni con regolarità (come report precompilati, set di dati, immagini statiche e modelli di accesso alle risorse) per identificare ridondanze, sottoutilizzi e obiettivi potenziali di disattivazione. Elimina le risorse ridondanti per ridurre gli sprechi nel tuo carico di lavoro.

Passaggi dell'implementazione

- Redigi l'inventario: esegui l'inventario completo per identificare tutte le risorse del tuo carico di lavoro.
- Analizza l'utilizzo: usa il monitoraggio continuo per identificare le risorse statiche che non sono più necessarie.
- Rimuovi le risorse inutilizzate: sviluppa un piano per rimuovere le risorse che non sono più necessarie.
  - Prima di rimuovere qualsiasi risorsa, valuta l'impatto della rimozione sull'architettura.
  - Analizza le risorse generate in sovrapposizione per rimuovere le elaborazioni ridondanti.
  - Aggiorna le tue applicazioni per smettere di produrre e archiviare risorse che non sono più necessarie.
- Comunica con le terze parti: istruisci le terze parti affinché smettano di produrre e archiviare per tuo conto risorse gestite non più necessarie. Chiedi di consolidare le risorse ridondanti.
- Usa le policy del ciclo di vita: utilizza le policy del ciclo di vita per eliminare automaticamente le risorse inutilizzate.
  - Puoi utilizzare il ciclo di vita Amazon S3 per gestire gli oggetti durante il loro ciclo di vita.
  - Puoi utilizzare Amazon Data Lifecycle Manager per automatizzare la creazione, la conservazione e l'eliminazione degli snapshot Amazon EBS e delle AMI basate su Amazon EBS.
- Rivedi e ottimizza: esamina regolarmente il carico di lavoro per identificare e rimuovere eventuali risorse inutilizzate.

Risorse

Documenti correlati:

- Ottimizzazione dell'infrastruttura AWS per la sostenibilità, Parte II: Achiviazione
- Come posso terminare risorse attive che non mi servono più sul mio Account AWS?

Video correlati:

- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Preserving and maximizing the value of digital media assets using Amazon S3
- AWS re:Invent 2023 - Optimize costs in your multi-account environments

SUS02-BP04 Ottimizzazione del posizionamento geografico dei carichi di lavoro in base ai requisiti di rete

Seleziona le sedi cloud e i servizi per il carico di lavoro per ridurre la distanza che il traffico di rete deve percorrere e diminuire così le risorse totali di rete richieste per supportare il carico di lavoro.

Anti-pattern comuni:

- Selezione della regione del carico di lavoro in base alla propria collocazione.
- Consolidamento di tutte le risorse del carico di lavoro in un'unica posizione geografica.
- Tutto il traffico passa attraverso i data center esistenti.

Vantaggi dell'adozione di questa best practice: il posizionamento di un carico di lavoro in prossimità dei relativi utenti garantisce la latenza più bassa e la contemporanea riduzione del trasferimento dei dati nella rete e dell'impatto ambientale.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

L'infrastruttura Cloud AWS viene definita con opzioni diverse relative alle sedi, come aree geografiche, zone di disponibilità, gruppi di posizionamento e posizioni edge come AWS Outposts e zone locali AWS. Queste opzioni relative alle sedi sono responsabili della gestione della connettività tra i componenti delle applicazioni, i servizi cloud, le reti edge e i data center on-premise.

Analizza i modelli di accesso alla rete nel tuo carico di lavoro per stabilire come usare queste opzioni relative alle sedi cloud e ridurre la distanza che il traffico di rete deve percorrere.

Passaggi dell'implementazione

- Analizza i modelli di accesso alla rete nel tuo carico di lavoro per capire come gli utenti usano la tua applicazione.
  - Usa strumenti di monitoraggio, come Amazon CloudWatch e AWS CloudTrail, per raccogliere i dati sull'attività della rete.
  - Analizza i dati per identificare il modello di accesso alla rete.
- Seleziona le regioni appropriate per l'implementazione del carico di lavoro in base ai seguenti elementi chiave:
  - Il tuo obiettivo di sostenibilità: spiegato nella Selezione dell'area geografica.
  - Dove si trovano i tuoi dati: per le applicazioni a uso intensivo di dati, ad esempio applicazioni di big data e machine learning, il codice dell'applicazione deve essere eseguito il più vicino possibile ai dati.
  - Dove si trovano i tuoi utenti: per le applicazioni rivolte agli utenti, scegli un'area geografica (o più di una) vicina agli utenti del tuo carico di lavoro.
  - Altre limitazioni: considera le limitazioni relative a costi e conformità, come spiegato in Cosa considerare quando si seleziona un'area geografica per i carichi di lavoro.
- Usa la cache locale o le Soluzioni per la acche di AWS per le risorse di frequente utilizzo per migliorare le performance, ridurre lo spostamento dei dati e minimizzare l'impatto ambientale.

Service	When to use
Amazon CloudFront	Usa per memorizzare nella cache contenuti statici come immagini, script e video, nonché contenuti dinamici come risposte API o applicazioni Web.
Amazon ElastiCache	Usa per memorizzare nella cache i contenuti per le applicazioni Web.
DynamoDB Accelerator	Usa per aggiungere accelerazione in memoria alle tabelle DynamoDB.

- Utilizza servizi in grado di supportarti nell'esecuzione del codice in posizioni più vicine agli utenti del carico di lavoro:

Service	When to use
Lambda@Edge	Usa per operazioni a uso intensivo di risorse di calcolo eseguite quando gli oggetti non si trovano nella cache.
Funzioni Amazon CloudFront	Usa per casi d'uso semplici, ad esempio manipolazioni di risposte o richieste HTTP(s) che possono essere avviate da funzioni di breve durata.
AWS IoT Greengrass	Usa per eseguire la memorizzazione nella cache di risorse di calcolo, messaggistica e dati per i dispositivi connessi.

- Utilizza il pooling delle connessioni per consentire il loro riutilizzo e ridurre le risorse richieste.
- Utilizza archivi di dati distribuiti che non si affidano a connessioni persistenti e aggiornamenti sincroni per garantire coerenza e servire le popolazioni regionali.
- Sostituisci la capacità di rete statica preassegnata con una capacità dinamica condivisa e condividi l'impatto in termini di sostenibilità della capacità di rete con altri sottoscrittori.

Risorse

Documenti correlati:

- Ottimizzazione dell'infrastruttura AWS per la sostenibilità, parte III: reti
- Documentazione Amazon ElastiCache
- Che cos'è Amazon CloudFront?
- Carateristiche principali di Amazon CloudFront
- Infrastruttura globale AWS
- Zone locali AWS e AWS Outposts, scelta della giusta tecnologia per un carico di lavoro edge
- Gruppi di collocazione
- Zone locali AWS
- AWS Outposts

Video correlati:

- Demistificazione del trasferimento dei dati su AWS
- Scaling network performance on next-gen Amazon EC2 instances
- Video di presentazione delle zone locali AWS
- AWS Outposts: Overview and How it Works
- AWS re:Invent 2023 - A migration strategy for edge and on-premises workloads
- AWS re:Invent 2021: AWS Outposts: Spostamento dell'esperienza AWS in un ambiente on-premise
- AWS re:Invent 2020 - AWS Wavelength: Run apps with ultra-low latency at 5G edge
- AWS re:Invent 2022: Zone locali AWS: creazione di applicazioni per una posizione edge distribuita
- AWS re:Invent 2021: Creazione di siti Web a bassa latenza con Amazon CloudFront
- AWS re:Invent 2022: Miglioramento delle prestazioni e della disponibilità con AWS Global Accelerator
- AWS re:Invent 2022: Creazione di una rete WAN usando AWS
- AWS re:Invent 2020: Gestione del traffico globale con Amazon Route 53

Esempi correlati:

- Workshop di rete AWS
- Progettazione di architetture per la sostenibilità: riduzione al minimo dello spostamento dei dati tra reti

SUS02-BP05 Ottimizzazione delle risorse dei membri del team in base alle attività eseguite

Ottimizza le risorse fornite ai membri del team per ridurre al minimo l'impatto sulla sostenibilità ambientale e supportare al tempo stesso le loro esigenze.

Anti-pattern comuni:

- Ignori l'impatto dei dispositivi utilizzati dai membri del tuo team sull'efficienza complessiva della tua applicazione cloud.
- Gestisci e aggiorni manualmente le risorse utilizzate dai membri del tuo team.

Vantaggi dell'adozione di questa best practice: se si ottimizzano le risorse dei membri del team si migliora l'efficienza generale delle applicazioni abilitate al cloud.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Identifica le risorse che i membri del tuo team usano per accedere ai tuoi servizi, il loro ciclo di vita atteso e l'impatto finanziario e di sostenibilità. Implementa strategie per ottimizzare queste risorse. Esegui ad esempio operazioni complesse, come rendering e compilazione, su infrastrutture scalabili altamente utilizzate, invece che su sistemi per utenti singoli, sottoutilizzati e con un alto dispendio energetico.

Passaggi dell'implementazione

- Usa workstation efficienti dal punto di vista energetico: fornisci ai membri del team workstation e periferiche efficienti dal punto di vista energetico. Utilizza in questi dispositivi funzionalità di gestione dell'alimentazione efficienti, come la modalità di risparmio energetico, per ridurne il consumo di energia.
- Usa la virtualizzazione: utilizza i desktop virtuali e lo streaming delle applicazioni per limitare gli aggiornamenti e i requisiti dei dispositivi.
- Favorisci la collaborazione remota: incoraggia i membri del team a utilizzare strumenti di collaborazione remota come Amazon Chime o AWS Wickr per ridurre la necessità di spostamenti aziendali e le emissioni di carbonio associate.
- Usa software efficienti dal punto di vista energetico: fornisci ai membri del team software efficienti dal punto di vista energetico rimuovendo o disattivando funzionalità e processi non necessari.
- Gestisci i cicli di vita: valuta l'impatto di processi e sistemi sul ciclo di vita dei tuoi dispositivi e seleziona le soluzioni che riducono al minimo la necessità di sostituzione dei dispositivi, pur continuando a soddisfare i requisiti di business. Effettua regolarmente la manutenzione e l'aggiornamento delle workstation o del software per conservare e migliorare l'efficienza.
- Usa la gestione remota dei dispositivi: implementa la gestione remota dei dispositivi per ridurre gli spostamenti aziendali.
  - Gestione dei gruppi di nodi AWS Systems Manager è un'esperienza di interfaccia utente unificata che ti aiuta a gestire da remoto i nodi in esecuzione su AWS oppure on-premises.

Risorse

Documenti correlati:

- Che cos'è Amazon WorkSpaces?
- Cost Optimizer per Amazon WorkSpaces
- Documentazione su Amazon AppStream 2.0
- NICE DCV

Video correlati:

- Gestire i costi per Amazon WorkSpaces su AWS

SUS02-BP06 Implementazione del buffering o della limitazione (della larghezza di banda della rete) per ridurre la curva della domanda

Il buffering e la limitazione (della larghezza di banda della rete) riducono la curva delle richieste e la capacità fornita tramite provisioning per il tuo carico di lavoro.

Anti-pattern comuni:

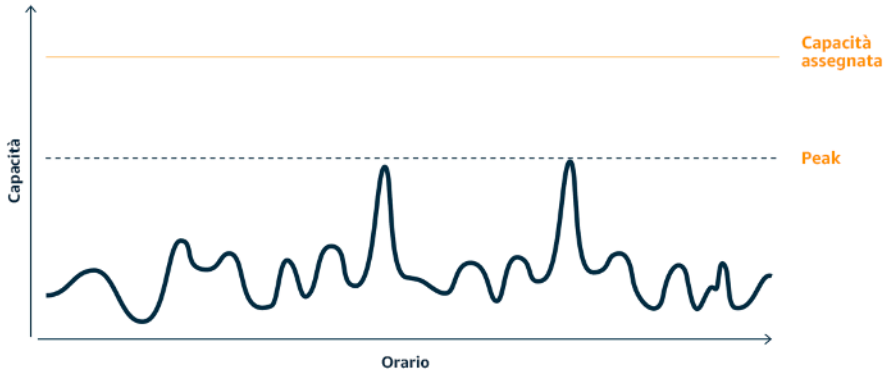
- Elabori immediatamente le richieste del client, anche se non è necessario.
- Non analizzi i requisiti relativi alle richieste dei clienti.

Vantaggi dell'azione di questa best practice: ridurre la curva della domanda per diminuire la capacità richiesta fornita tramite provisioning per il carico di lavoro. Ridurre la capacità fornita tramite provisioning significa ridurre il consumo di energia e contenere l'impatto ambientale.

Livello di rischio associato alla mancata adozione di questa best practice: basso

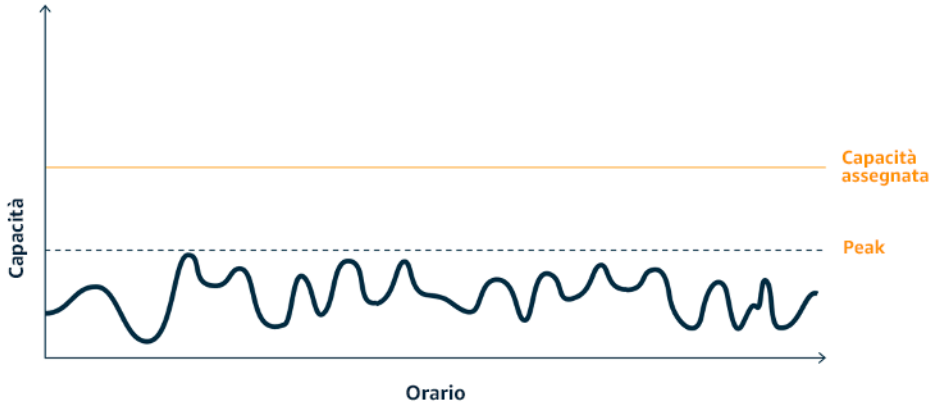
Guida all'implementazione

Diminuire la curva della domanda del carico di lavoro può aiutarti a ridurre la capacità fornita tramite provisioning di un carico di lavoro, oltre al suo impatto sull'ambiente. Supponiamo che un carico di lavoro abbia la curva della domanda mostrata nella figura qui sotto. Questo carico di lavoro presenta due picchi e per gestire tali picchi viene eseguito il provisioning della capacità di risorse mostrata dalla linea arancione. Le risorse e l'energia utilizzate per questo carico di lavoro non sono indicate nell'area sotto la curva della domanda, ma nell'area sotto la linea della capacità fornita, poiché per gestire questi due picchi è necessario eseguire il provisioning di tale capacità.



Curva della domanda con due picchi distinti che richiedono una capacità elevata.

Puoi usare il buffering o la limitazione (della larghezza di banda della rete) per modificare la curva della domanda e appianare i picchi, con conseguente diminuzione della capacità fornita tramite provisioning e consumo inferiore di energia. Implementa la limitazione (della larghezza di banda della rete) quando i client eseguono nuovi tentativi. Implementa il buffering per archiviare la richiesta e rinviare l'elaborazione a un secondo momento.



L'effetto della limitazione (della larghezza di banda della rete) sulla curva della domanda e la capacità fornita tramite provisioning.

Passaggi dell'implementazione

- Analizza le richieste del client per stabilire come rispondere. Le domande da considerare includono:
  - Questa richiesta può essere elaborata in modo asincrono?

    - Il client ha la possibilità di ripetere i tentativi?
- Se il client ha la possibilità di ripetere i tentativi puoi implementare la limitazione (della larghezza di banda della rete), che indica alla sorgente che, se non è in grado di soddisfare la richiesta all'ora corrente, dovrebbe riprovare più tardi.
    - Puoi usare Amazon API Gateway per implementare la limitazione (della larghezza di banda della rete).
- Per i client che non possono eseguire altri tentativi, è necessario implementare un buffer per ridurre i picchi della curva della domanda. Il buffering rinvia l'elaborazione delle richieste, consentendo alle applicazioni eseguite a velocità diverse di comunicare in modo efficace. Un approccio basato sul buffering impiega una coda o un flusso per l'accettazione dei messaggi dai produttori. I messaggi vengono letti ed elaborati dai consumatori e ciò consente ai messaggi di essere eseguiti alla velocità che soddisfa i requisiti aziendali del consumatore stesso.
    - Amazon Simple Queue Service(Amazon SQS) è un servizio gestito che offre code che consentono a un singolo consumatore di leggere singoli messaggi.
    - Amazon Kinesis offre un flusso che consente a più consumatori di leggere gli stessi messaggi.
- Analizza la domanda complessiva, la velocità di modifica e il tempo di risposta richiesto per determinare le dimensioni del throttling o del buffer richiesto.

Risorse

Documenti correlati:

- Nozioni di base su Amazon SQS
- Integrazione dell'applicazione con code e messaggi
- Gestione e monitoraggio della limitazione delle API nei carichi di lavoro
- Throttling a tiered, multi-tenant REST API at scale using API Gateway
- Integrazione dell'applicazione con code e messaggi

Video correlati:

- AWS re:Invent 2022 - Application integration patterns for microservices
- AWS re:Invent 2023 - Smart savings: Amazon EC2 cost-optimization strategies
- AWS re:Invent 2023 - Advanced integration patterns & trade-offs for loosely coupled systems

Software e architettura

Domanda

- SUS 3 In che modo sfrutti i modelli di software e architetture per sostenere i tuoi obiettivi di sostenibilità?

SUS 3 In che modo sfrutti i modelli di software e architetture per sostenere i tuoi obiettivi di sostenibilità?

Implementa modelli per eseguire lo smoothing del carico e garantire un utilizzo elevato e coerente delle risorse implementate per ridurre al minimo il loro consumo. In seguito alle modifiche nei comportamenti degli utenti nel tempo, alcuni componenti potrebbero diventare inattivi per mancanza di utilizzo. Rivedi modelli e architetture per consolidare i componenti sottoutilizzati e aumentare l'uso complessivo. Ritira i componenti che non sono più necessari. Analizza le prestazioni dei componenti dei tuoi carichi di lavoro e ottimizza quelli che usano la maggior quantità di risorse. Identifica i dispositivi che i clienti utilizzano per accedere ai servizi e implementa modelli in grado di ridurre al minimo la necessità di aggiornamenti dei dispositivi.

Best practice

- SUS03-BP01 Ottimizzazione del software e delle architetture per processi asincroni e pianificati
- SUS03-BP02 Rimozione o rifattorizzazione dei componenti dei carichi di lavoro con un utilizzo ridotto o assente
- Ottimizzazione delle aree di codice che consumano la maggior parte del tempo o delle risorse
- SUS03-BP04 Ottimizzazione dell'impatto su dispositivi e apparecchiature
- SUS03-BP05 Uso dei modelli e le architetture software che meglio supportano l'accesso ai dati e i modelli di archiviazione

SUS03-BP01 Ottimizzazione del software e delle architetture per processi asincroni e pianificati

Utilizza modelli efficienti di software e di architettura, come quelli basati sulle code, per mantenere un utilizzo elevato e costante delle risorse distribuite.

Anti-pattern comuni:

- Provisioning di risorse in eccedenza per il carico di lavoro in cloud con lo scopo di far fronte a picchi di domanda imprevisti.

- Architettura non in grado di disaccoppiare i mittenti e i ricevitori di messaggi asincroni mediante un componente di messaggistica.

Vantaggi dell'adozione di questa best practice:

- Modelli efficienti di software e architettura riducono al minimo le risorse inutilizzate nel carico di lavoro e migliorano l'efficienza complessiva.
- È possibile dimensionare le risorse dedicate all'elaborazione indipendentemente dalla ricezione di messaggi asincroni.
- Grazie a un componente di messaggistica, i requisiti di disponibilità si attenuano e possono essere soddisfatti con un numero inferiore di risorse.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Utilizza modelli di architettura efficienti, come l'architettura basata su eventi, che consentono un utilizzo uniforme dei componenti e riducono al minimo il provisioning in eccedenza nel carico di lavoro. L'utilizzo di modelli architetturali efficienti riduce al minimo le risorse inattive a causa del mancato utilizzo dovuto alle variazioni della domanda nel tempo.

Comprendi i requisiti dei componenti del carico di lavoro e adotta modelli di architettura che aumentino l'utilizzo complessivo delle risorse. Ritira i componenti che non sono più necessari.

Passaggi dell'implementazione

- Analizza le esigenze del tuo carico di lavoro per determinare come rispondere a tali richieste.
- Per le richieste o i processi che non necessitano di risposte sincrone, utilizza architetture basate su code e worker a scalabilità automatica per massimizzare l'utilizzo. Ecco alcuni esempi in cui potresti prendere in considerazione un'architettura basata sulle code:

Queuing mechanism	Description
code di processi AWS Batch	I processi AWS Batch vengono inviati a una coda di processi, dove risiedono fino a quando non è possibile pianificare la loro esecuzione in un ambiente di calcolo.

Queuing mechanism	Description
Amazon Simple Queue Service e istanze spot Amazon EC2	Abbina Amazon SQS e istanze spot per realizzare un'architettura efficiente e in grado di tollerare i malfunzionamenti.

- Per le richieste o i processi che possono essere elaborati in qualsiasi momento, ottieni una maggiore efficienza utilizzando i meccanismi di pianificazione dell'elaborazione delle attività in blocco. Ecco alcuni esempi di meccanismi di pianificazione su AWS:

Scheduling mechanism	Description
Sistema di pianificazione Amazon EventBridge	Una funzionalità di Amazon EventBridge che consente di creare, eseguire e gestire attività pianificate su larga scala.
Pianificazione basata sul tempo di AWS Glue	Definisci una pianificazione in base al tempo per crawler e processi in AWS Glue.
Attività pianificate di Amazon Elastic Container Service (Amazon ECS)	Amazon ECS supporta le creazione di attività pianificate. Le attività pianificate utilizzano le regole di Amazon EventBridge per l'esecuzi one in base a una pianificazione p in risposta a un evento EventBridge.
Instance Scheduler	Configura le pianificazioni di avvio e arresto delle istanze Amazon EC2 e Amazon Relational Database Service.

- Se nella tua architettura utilizzi meccanismi di polling e webhook, sostituiscili con eventi. Utilizza architetture basate su eventi per realizzare carichi di lavoro efficienti.
- Approfitta dei servizi serverless su AWS per eliminare la necessità di provisioning in eccedenza sull'infrastruttura.
- Dimensiona in modo appropriato i singoli componenti dell'architettura per evitare la presenza di risorse inattive in attesa di input.
  - Puoi utilizzare i suggerimenti per il dimensionamento appropriato in AWS Cost Explorer o AWS Compute Optimizer per identificare le corrette opportunità di dimensionamento.

- Per maggiori dettagli, consulta Right Sizing: Provisioning Instances to Match Workloads.

Risorse

Documenti correlati:

- Che cos'è Amazon Simple Queue Service?
- Che cos'è Amazon MQ?
- Dimensionamento basato su Amazon SQS
- Che cos'è AWS Step Functions?
- Che cos'è AWS Lambda?
- Utilizzo di AWS Lambda con Amazon SQS
- Che cos'è Amazon EventBridge?
- Managing Asynchronous Workflows with a REST API

Video correlati:

- AWS re:Invent 2023 - Navigating the journey to serverless event-driven architecture
- AWS re:Invent 2023 - Using serverless for event-driven architecture & domain-driven design
- AWS re:Invent 2023 - Advanced event-driven patterns with Amazon EventBridge
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- Asynchronous Message Patterns | AWS Events

Esempi correlati:

- Event-driven architecture with AWS Graviton Processors and Amazon EC2 Spot Instances

SUS03-BP02 Rimozione o rifattorizzazione dei componenti dei carichi di lavoro con un utilizzo ridotto o assente

Elimina i componenti non utilizzati e non più necessari e rifattorizza quelli con scarso utilizzo per limitare lo spreco di risorse nel tuo carico di lavoro.

Anti-pattern comuni:

- Non verifichi con regolarità il livello di utilizzo dei singoli componenti del tuo carico di lavoro.
- Non verifichi e analizzi i consigli ricevuti dagli strumenti di dimensionamento AWS, ad esempio AWS Compute Optimizer.

Vantaggi dell'adozione di questa best practice: se si eliminano i componenti non utilizzati si riducono gli sprechi e si migliora l'efficienza generale del carico di lavoro cloud.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Esamina il tuo carico di lavoro per identificare componenti inattivi o non utilizzati. Si tratta di un processo di miglioramento iterativo che può essere attivato da cambiamenti della domanda o dal rilascio di un nuovo servizio cloud. Ad esempio, una riduzione significativa del runtime delle funzioni di AWS Lambda può essere un indicatore della necessità di diminuire la dimensione della memoria. Inoltre, quando AWS rilascia nuovi servizi e funzionalità, è possibile che i servizi ottimali e l'architettura per il carico di lavoro cambino.

Monitora continuamente l'attività del carico di lavoro e cerca le opportunità per migliorare il livello di utilizzo dei singoli componenti. Eliminando i componenti inattivi ed eseguendo attività di ridimensionamento, soddisfi i requisiti aziendali con il numero minimo di risorse cloud.

Passaggi dell'implementazione

- Redigi l'inventario delle tue risorse AWS. In AWS, puoi attivare Esploratore di risorse AWS per esplorare e organizzare le tue risorse AWS. Per maggiori dettagli, consulta AWS re:Invent 2022 - How to manage resources and applications at scale on AWS.
- Monitora e acquisici metriche di utilizzo per componenti strategici del tuo carico di lavoro (like l'utilizzo della CPU, l'utilizzo della memoria o la velocità di trasmissione effettiva nelle metriche Amazon CloudWatch).
- Individua i componenti inutilizzati o sottoutilizzati nell'architettura.
  - Per carichi di lavoro stabili, verifica gli strumenti di ridimensionamento AWS come AWS Compute Optimizer a intervalli regolari per individuare componenti inattivi, inutilizzati o sottoutilizzati.
  - Per carichi di lavoro effimeri, valuta metriche di utilizzo per identificare componenti inattivi, inutilizzati o sottoutilizzati.
- Ritira componenti e risorse associate (come le immagini Amazon ECR) che non sono più necessarie.

  - Pulizia automatica delle immagini inutilizzate in Amazon ECR
  - Elimina volumi Amazon Elastic Block Store (Amazon EBS) utilizzando AWS Config e AWS Systems Manager
- Rifattorizza o consolida i componenti sottoutilizzati con altre risorse per promuovere un utilizzo efficiente. Ad esempio, puoi eseguire il provisioning di più database di piccole dimensioni su una singola istanza di database Amazon RDS invece di eseguire database su singole istanze sottoutilizzate.
- Scopri le risorse fornite dal tuo carico di lavoro per completare un'unità di lavoro.

Risorse

Documenti correlati:

- AWS Trusted Advisor
- Che cos'è Amazon CloudWatch?
- Ridimensionamento corretto: provisioning delle istanze per soddisfare i carichi di lavoro
- Optimizing your cost with Rightsizing Recommendations

Video correlati:

- AWS re:Invent 2023 - Capacity, availability, cost efficiency: Pick three

Esempi correlati:

- Optimize Hardware Patterns and Observe Sustainability KPIs

Ottimizzazione delle aree di codice che consumano la maggior parte del tempo o delle risorse

Ottimizza il codice eseguito all'interno di diversi componenti della tua architettura per ridurre l'utilizzo delle risorse e massimizzare al tempo stesso le prestazioni.

Anti-pattern comuni:

- Ignori l'ottimizzazione del codice per l'utilizzo delle risorse.
- In genere, rispondi ai problemi di performance aumentando le risorse.

- La revisione del codice e il processo di sviluppo non monitorano le modifiche a livello di performance.

Vantaggi dell'adozione di questa best practice: l'uso di un codice efficiente riduce la quantità di risorse utilizzate e migliora le performance.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

È fondamentale esaminare ogni area funzionale, incluso il codice per un'applicazione ideata nel cloud, per ottimizzare l'uso delle risorse e le performance. Monitora costantemente le performance del tuo carico di lavoro negli ambienti di sviluppo e produzione e identifica le opportunità per migliorare gli snippet di codice che comportano un utilizzo particolarmente elevato delle risorse. Adotta un processo di revisione con cadenza regolare per identificare i bug o gli anti-pattern all'interno del codice che utilizzano le risorse in modo non efficiente. Sfrutta algoritmi semplici ed efficienti che hanno gli stessi risultati per il tuo caso d'uso.

Passaggi dell'implementazione

- Usa un linguaggio di programmazione efficiente: utilizza un sistema operativo e un linguaggio di programmazione efficienti per il carico di lavoro. Per dettagli sui linguaggi di programmazione efficienti dal punto di vista delle risorse (incluso Rust), consulta Sostenibilità con Rust.
- Usa un assistente per la scrittura del codice basato sull'IA: prendi in considerazione l'utilizzo di un assistente per la scrittura del codice basato sull'IA, ad esempio Amazon CodeWhisperer, per scrivere il codice in modo efficiente.
- Automatizza le revisioni del codice: mentre sviluppi i tuoi carichi di lavoro, adotta un processo di revisione del codice automatizzato, per migliorare la qualità e identificare bug e anti-pattern.
  - Automazione delle revisioni del codice con il Amazon CodeGuru Reviewer
  - Rilevare i bug concomitanti con Amazon CodeGuru
  - Aumentare la qualità del codice per le applicazioni Python con Amazon CodeGuru
- Usa un profiler di codice: utilizza un profiler di codice per identificare le aree che impiegano più tempo o risorse e trasformale in obiettivi di ottimizzazione.
  - Ridurre l'impatto ambientale della tua organizzazione con Amazon CodeGuru Profiler
  - Capire l'utilizzo della memoria nella tua applicazione Java con Amazon CodeGuru Profiler
  - Migliorare l'esperienza del cliente e ridurre i costi con Amazon CodeGuru Profiler

- Monitora e ottimizza: utilizza le risorse di monitoraggio continuo per identificare i componenti con requisiti di risorse elevati o una configurazione non ottimale.
  - Sostituisci gli algoritmi a uso intensivo di elaborazioni con una versione più semplice ed efficiente che produce gli stessi risultati.
  - Rimuovi il codice non necessario, come quello relativo all'ordinamento e alla formattazione.
- Usa la rifattorizzazione o la trasformazione del codice: scopri la capacità di trasformazione del codice di Amazon Q per eseguire la manutenzione e gli aggiornamenti delle applicazioni.
  - Aggiorna le versioni linguistiche con la trasformazione del codice di Amazon Q
  - AWS re:Invent 2023 - Automate app upgrades & maintenance using Amazon Q Code Transformation

Risorse

Documenti correlati:

- Che cos'è Amazon CodeGuru Profiler?
- Istanze FPGA
- SDK AWS su Strumenti per creare su AWS

Video correlati:

- Migliora l'efficienza del codice con Amazon CodeGuru Profiler
- AWS re:Invent 2023 - Best practices for Amazon CodeWhisperer
- Automatizza le revisioni del codice e i consigli sulle prestazioni dell'applicazione con Amazon CodeGuru

Esempi correlati:

- Optimizing Code with Amazon CodeGuru

SUS03-BP04 Ottimizzazione dell'impatto su dispositivi e apparecchiature

Conoscere i dispositivi e le apparecchiature utilizzate nell'architettura e applicare strategie per ridurre il loro uso. Questo può ridurre l'impatto ambientale complessivo del tuo carico di lavoro cloud.

Anti-pattern comuni:

- Ignori l'impatto ambientale dei dispositivi utilizzati dai clienti.
- Gestisci e aggiorni manualmente le risorse utilizzate dai clienti.

Vantaggi dell'adozione di questa best practice: implementare modelli e funzionalità software ottimizzati per i dispositivi dei clienti può ridurre l'impatto ambientale complessivo del carico di lavoro del cloud.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Implementare modelli e funzionalità software ottimizzati per i dispositivi dei clienti può ridurre l'impatto ambientale in diversi modi:

- Implementare nuove funzionalità compatibili con le versioni precedenti può ridurre il numero di sostituzioni hardware.
- Ottimizzare un'applicazione per un'esecuzione efficiente sui dispositivi può contribuire a ridurre l'utilizzo di energia ed estendere la durata della loro batteria (se sono alimentati tramite batteria).
- Ottimizzare un'applicazione per i dispositivi significa anche ridurre il trasferimento dei dati sulla rete.

Conoscere i dispositivi e l'attrezzatura utilizzati nella tua architettura, il loro ciclo di vita atteso e l'impatto della sostituzione di tali componenti. Implementare modelli e funzionalità software che possono contribuire a ridurre l'uso di energia da parte del dispositivo, la necessità da parte dei clienti di sostituirlo e anche di eseguire l'aggiornamento manuale.

Passaggi dell'implementazione

- Redigi l'inventario: esegui l'inventario dei dispositivi utilizzati nell'architettura. I dispositivi possono essere cellulari, tablet, dispositivi IOT, illuminazione smart o persino dispositivi smart in una fabbrica.
- Usa dispositivi efficienti dal punto di vista energetico: prendi in considerazione l'utilizzo di dispositivi ad alta efficienza energetica nell'architettura. Utilizza le configurazioni di gestione dell'alimentazione sui dispositivi per accedere alla modalità di risparmio energetico quando non sono in uso.
- Esegui applicazioni efficienti: ottimizza l'applicazione in esecuzione sui dispositivi:
  - Usa strategie come l'esecuzione di attività in background per ridurre l'uso di energia.

  - Prendi in considerazione la larghezza di banda e la latenza della rete durante la creazione di payload e implementa funzionalità che consentano alle tue applicazioni di lavorare bene anche in presenza di una larghezza di banda ridotta e di link ad alta latenza.
  - Converti payload e file in formati ottimizzati richiesti dai dispositivi. Ad esempio, puoi usare Amazon Elastic Transcoder o AWS Elemental MediaConvert per convertire file di media digitali di grandi dimensioni e di qualità elevata in formati che gli utenti possono riprodurre su dispositivi mobili, tablet, browser web e televisioni connesse.
  - Esegui attività a elevata intensità computazionale lato server (come, ad esempio, il rendering delle immagini) oppure usa lo streaming delle applicazioni per migliorare l'esperienza utente sui dispositivi di versioni precedenti.
  - Esegui la segmentazione e la paginazione dell'output, soprattutto per le sessioni interattive, per gestire i payload e limitare i requisiti di archiviazione in locale.
- Coinvolgi i fornitori: collabora con i fornitori di dispositivi che utilizzano materiali sostenibili e forniscono trasparenza nelle catene di approvvigionamento e nelle certificazioni ambientali.
- Usa gli aggiornamenti via etere (OTA): utilizza il meccanismo automatico via etere (OTA) per implementare gli aggiornamenti su uno o più dispositivi.
  - Puoi usare una pipeline CI/CD per aggiornare le applicazioni mobili.
  - Puoi usare AWS IoT Device Management per gestire da remoto dispositivi connessi su scala.
- Usa device farm gestite: per testare nuove funzionalità e aggiornamenti, utilizza device farm gestite con set di hardware rappresentativi e itera lo sviluppo per ottimizzare i dispositivi supportati. Per ulteriori dettagli, consulta SUS06-BP04 Utilizzo di device farm gestite per i test.
- Monitora e migliora in modo continuo: monitora il consumo energetico dei dispositivi per identificare le aree di miglioramento. Utilizza le nuove tecnologie o best practice per migliorare l'impatto ambientale di questi dispositivi.

Risorse

Documenti correlati:

- What is AWS Device Farm?
- AppStream 2.0 Documentation
- NICE DCV
- Tutorial OTA per l'aggiornamento del firmware su dispositivi che eseguono FreeRTOS
- Optimizing Your IoT Devices for Environmental Sustainability

Video correlati:

- AWS re:Invent 2023 - Improve your mobile and web app quality using AWS Device Farm

SUS03-BP05 Uso dei modelli e le architetture software che meglio supportano l'accesso ai dati e i modelli di archiviazione

Scopri come i dati vengono utilizzati all'interno del tuo carico di lavoro, consumati dagli utenti, trasferiti e archiviati. Usa architetture e modelli software in grado di supportare al meglio l'accesso ai dati e l'archiviazione per ridurre le risorse di elaborazione, rete e storage richieste dal carico di lavoro.

Anti-pattern comuni:

- Ritieni che tutti i carichi di lavoro abbiano modelli di accesso e archiviazione dei dati simili.
- Utilizzi un solo livello di archiviazione, presupponendo che tutti i carichi di lavoro rientrino in tale livello.
- Ritieni che gli schemi di accesso ai dati rimarranno coerenti nel tempo.
- La tua architettura supporta una potenziale espansione elevata dell'accesso ai dati, con conseguente inattività delle risorse per la maggior parte del tempo.

Vantaggi dell'adozione di questa best practice: selezionando e ottimizzando la tua architettura in base all'accesso ai dati e ai modelli di archiviazione diminuirà la complessità dello sviluppo e aumenterà l'utilizzo complessivo. Capire quando utilizzare le tabelle globali, il partizionamento dei dati e la memorizzazione nella cache, ti aiuterà a ridurre i costi operativi e a effettuare il dimensionamento in base alle esigenze del carico di lavoro.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Usa modelli di software e architetture che siano quanto più in linea con le caratteristiche dei tuoi dati e i modelli di accesso. Ad esempio, usa un'architettura di dati moderni su AWS che ti consenta di utilizzare servizi dedicati ottimizzati per i tuoi casi d'uso di analisi specifici. Questi modelli di architettura consentono un'elaborazione efficiente dei dati e riducono l'utilizzo delle risorse.

Passaggi dell'implementazione

- Analizza le caratteristiche dei dati e i modelli di accesso per individuare la configurazione corretta per le tue risorse cloud. Gli aspetti chiave da considerare includono:

  - Tipi di dati: strutturati, semi-strutturati, non strutturati
  - Crescita dei dati: delimitati, non delimitati
  - Durabilità dei dati: persistenti, effimeri, transitori
  - Modelli di accesso: letture o scritture, frequenza di aggiornamento, con picchi o costante
- Usa tipi di architetture che meglio supportano l'accesso ai dati e i modelli di archiviazione.
  - Pattern per consentire la persistenza dei dati
  - Progettiamo! Architetture dei dati moderne
  - Database su AWS: lo strumento più adatto per ciascun processo
- Sfrutta le tecnologie che lavorano in modo nativo con i dati compressi.
  - Formati file di supporto alla compressione di Athena
  - Opzioni di formato per input e output ETL in AWS Glue
  - Caricamento di file di dati compressi da Amazon S3 con Amazon Redshift
- Usa servizi di analisi per l'elaborazione dei dati nella tua architettura. Per informazioni dettagliate sui servizi di analisi personalizzati di AWS, consulta AWS re:Invent 2022 - Building modern data architectures on AWS.
- Utilizza il motore del database che meglio supporta il modello di query dominante. Gestisci gli indici di database per garantire un'esecuzione efficiente delle query. Per ulteriori dettagli, consulta Database AWS e AWS re:Invent 2022 - Modernize apps with purpose-built databases.
- Seleziona protocolli di rete che riducano la quantità di capacità di rete utilizzata dalla tua architettura.

Risorse

Documenti correlati:

- COPY dai formati dei dati in colonne con Amazon Redshift
- Convertire il formato dei record di input in Firehose
- Migliora le prestazioni delle query su Amazon Athena con una conversione ai formati in colonne
- Monitoraggio del carico del database con Performance Insights su Amazon Aurora
- Monitoraggio del carico del database con Performance Insights su Amazon RDS
- Classe di storage Amazon S3 Intelligent-Tiering
- Build a CQRS event store with Amazon DynamoDB

Video correlati:

- AWS re:Invent 2022 - Building data mesh architectures on AWS
- AWS re:Invent 2023 - Deep dive into Amazon Aurora and its innovations
- AWS re:Invent 2023 - Improve Amazon EBS efficiency and be more cost-efficient
- AWS re:Invent 2023 - Optimizing storage price and performance with Amazon S3
- AWS re:Invent 2023 - Building and optimizing a data lake on Amazon S3
- AWS re:Invent 2023 - Advanced event-driven patterns with Amazon EventBridge

Esempi correlati:

- AWS Purpose Built Databases Workshop
- AWS Modern Data Architecture Immersion Day
- Build a Data Mesh on AWS

Dati

Domanda

- SUS 4 Come si può usufruire delle policy e dei modelli di gestione dei dati per supportare gli obiettivi di sostenibilità?

SUS 4 Come si può usufruire delle policy e dei modelli di gestione dei dati per supportare gli obiettivi di sostenibilità?

Implementa procedure di gestione dei dati per ridurre l'archiviazione assegnata richiesta per supportare il carico di lavoro e le risorse necessarie per l'uso correlato. Analizza i tuoi dati e usa tecnologie e configurazioni di archiviazione che supportano più efficacemente il valore aziendale dei dati e il modo in cui vengono utilizzati. Esegui il ciclo di vita dei dati su un'archiviazione più efficiente e meno performante al diminuire dei requisiti ed elimina i dati che non sono più necessari.

Best practice

- SUS04-BP01 Implementazione di una policy di classificazione dei dati
- SUS04-BP02 Utilizzo di tecnologie che supportano l'accesso ai dati e i modelli di archiviazione
- SUS04-BP03 Utilizzo delle policy per gestire il ciclo di vita dei set di dati

- SUS04-BP04 Utilizzo dell'elasticità e dell'automazione per espandere lo storage a blocchi o il file system
- SUS04-BP05 Eliminazione dei dati ridondanti o non necessari
- SUS04-BP06 Utilizzo di file system condivisi o archiviazione per accedere a dati comuni
- SUS04-BP07 Riduzione al minimo dello spostamento di dati tra reti
- SUS04-BP08 Backup dei dati solo quando sono difficili da ricreare

SUS04-BP01 Implementazione di una policy di classificazione dei dati

Classifica i dati per capire le criticità rispetto ai risultati aziendali e scegli il livello di archiviazione ad alta efficienza corretto per le tue informazioni.

Anti-pattern comuni:

- Non identifichi asset di dati con caratteristiche simili (come sensibilità, criticità aziendale o requisiti normativi) che vengono elaborati o archiviati.
- Non hai implementato un catalogo di dati per eseguire l'inventario dei tuoi asset.

Vantaggi derivanti dall'adozione di questa best practice: l'implementazione di una policy di classificazione dei dati ti consente di stabilire il livello di archiviazione per i dati più efficiente dal punto di vista energetico.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

La classificazione dei dati comporta l'identificazione dei tipi di dati elaborati e archiviati in un sistema informativo di proprietà o gestito da un'organizzazione. Inoltre, è necessario stabilire la criticità dei dati e il probabile impatto di una compromissione, perdita o uso improprio dei dati.

Implementare la policy di classificazione dei dati partendo dall'uso contestuale dei dati e creando uno schema di categorizzazione che tenga conto del livello di criticità di un determinato set di dati per le operazioni dell'organizzazione.

Passaggi dell'implementazione

- Esegui l'inventario dei dati: redigi l'inventario dei vari tipi di dati esistenti per il carico di lavoro.
- Raggruppa i dati: determina la criticità, la riservatezza, l'integrità e la disponibilità dei dati in base al rischio per l'organizzazione. Utilizza questi requisiti per raggruppare i dati in uno dei livelli di

classificazione dei dati adottati. Come esempio vedi Quattro semplici passaggi per classificare i tuoi dati e proteggere la tua startup.

- Definisci i livelli di classificazione dei dati e le policy: per ogni gruppo di dati, definisci il livello di classificazione dei dati, ad esempio pubblico o riservato, e le policy di gestione. Applica ai dati i tag adeguati. Per maggiori dettagli sulle categorie di classificazione dei dati, consulta il whitepaper sulla classificazione dei dati.
- Rivedi periodicamente: esamina e controlla periodicamente il tuo ambiente per individuare i dati senza tag e non classificati. Usa l'automazione per identificare questi dati, classificandoli e applicando i tag in modo appropriato. Come esempio vedi Catalogo dati e crawler in AWS Glue.
- Crea un catalogo dati: definisci un catalogo dati che fornisca funzionalità di audit e governance.
- Documenta: crea i documenti per comunicare le policy di classificazione dei dati e le procedure di gestione per ogni classe di dati.

Risorse

Documenti correlati:

- Utilizzo di Cloud AWS per supportare la classificazione dei dati
- Policy di tag di AWS Organizations

Video correlati:

- AWS re:Invent 2022 - Enabling agility with data governance on AWS
- AWS re:Invent 2023 - Data protection and resilience with AWS storage

SUS04-BP02 Utilizzo di tecnologie che supportano l'accesso ai dati e i modelli di archiviazione

Usa tecnologie di archiviazione in grado di supportare al meglio il modo in cui viene effettuato l'accesso ai dati e come vengono archiviati per ridurre la quantità di risorse assegnate e supportare al tempo stesso il tuo carico di lavoro.

Anti-pattern comuni:

- Ritieni che tutti i carichi di lavoro abbiano modelli di accesso e archiviazione dei dati simili.
- Utilizzi un solo livello di archiviazione, presupponendo che tutti i carichi di lavoro rientrino in tale livello.

- Ritieni che gli schemi di accesso ai dati rimarranno coerenti nel tempo.

Vantaggi derivanti da questa best practice: selezionare e ottimizzare le tecnologie di archiviazione in base all'accesso ai dati e ai modelli di archiviazione ti consentirà di ridurre le risorse cloud richieste per soddisfare le tue esigenze aziendali e migliorare l'efficienza generale del tuo carico di lavoro cloud.

Livello di rischio associato alla mancata adozione di questa best practice: basso

Guida all'implementazione

Seleziona la soluzione di archiviazione più adatta ai tuoi modelli di accesso. In alternativa, puoi modificarli affinché siano in linea con la soluzione di archiviazione, allo scopo di ottimizzare l'efficienza delle prestazioni.

Passaggi dell'implementazione

- Esamina le caratteristiche dei dati e dell'accesso: valuta le caratteristiche dei dati e il modello di accesso per raccogliere gli aspetti chiave delle tue esigenze di archiviazione. Gli aspetti chiave da considerare includono:
  - Tipi di dati: strutturati, semi-strutturati, non strutturati
  - Crescita dei dati: delimitati, non delimitati
  - Durabilità dei dati: persistenti, effimeri, transitori
  - Modelli di accesso: letture o scritture, frequenza, con picchi o costante
- Scegli la giusta tecnologia di archiviazione: migra i dati alla tecnologia di archiviazione appropriata che supporta le caratteristiche dei dati e il modello di accesso che usi. Ecco alcuni esempi di tecnologie di archiviazione AWS e delle loro caratteristiche chiave:

Type	Technology	Key characteristics
Archiviazione di oggetti	Amazon S3	Un servizio di archiviazione di oggetti con scalabilità illimitata, elevata disponibilità e più opzioni di accessibilità. Il trasferimento di oggetti e l'accesso a oggetti in e fuori da Amazon S3 possono

Type	Technology	Key characteristics
		utilizzare un servizio, ad esempio Transfer Acceleration o Access Points per supportare la tua posizione, le esigenze di sicurezza e i modelli di accesso.
Archiviazione	Amazon S3 Glacier	Classe di archiviazione di Amazon S3 creata per l'archiviazione dei dati.
File system condiviso	Amazon Elastic File System (Amazon EFS)	File system montabile a cui possono accedere più tipi di soluzioni di calcolo. Amazon EFS aumenta o riduce automaticamente lo spazio di archiviazione, mentre le relative prestazioni sono ottimizzate in modo da offrire costantemente latenze basse.

Type	Technology	Key characteristics
File system condiviso	Amazon FSx	Sviluppato sulle più recenti soluzioni di elaborazione AWS per supportare quattro file system comunemente usati: NetApp ONTAP, OpenZFS, Windows File Server e Lustre. Amazon FSx latenza, throughput e IOPS variano per file system e dovrebbero essere presi in considerazione quando selezioni il file system corretto per le esigenze del tuo carico di lavoro.
Archiviazione a blocchi	Amazon Elastic Block Store (Amazon EBS)	Servizio di archiviazione a blocchi scalabile e ad alte prestazioni progettato per Amazon Elastic Compute Cloud (Amazon EC2). Amazon EBS include l'archiviazione supportata da SSD per carichi di lavoro transazionali e a uso intensivo di IOPS, nonché archiviazione supportata da HDD per carichi di lavoro con uso intensivo della velocità di trasmissione effettiva.

Type	Technology	Key characteristics
Database relazionale	Amazon Aurora, Amazon RDS, Amazon Redshift	Progettati per supportare le transazioni ACID (atomicità, coerenza, isolamento, durabilità) e per mantenere l'integrità referenziale e una solida coerenza dei dati. Molte applicazioni tradizionali e sistemi Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) ed e-commerce utilizzano database relazionali per archiviare i propri dati.
Database chiave-valore	Amazon DynamoDB	Ottimizzati per schemi di accesso di uso comune, in genere per archiviare e recuperare grandi volumi di dati. Le app Web dal traffico elevato, i sistemi di e-commerce e le applicazioni di videogiochi sono casi d'uso tipici dei database chiave-valore.

- Automatizza l'allocazione dello spazio di archiviazione: per i sistemi di archiviazione con dimensione fissa, come Amazon EBS o Amazon FSx, monitora lo spazio di archiviazione disponibile e automatizza l'allocazione dell'archiviazione al raggiungimento di una soglia. È possibile sfruttare Amazon CloudWatch al fine di raccogliere e analizzare metriche diverse per Amazon EBS e Amazon FSx.
- Scegli la classe di archiviazione corretta: seleziona la classe di archiviazione appropriata per i tuoi dati.
  - Le classi di archiviazione Amazon S3 possono essere configurate a livello di oggetto. Un singolo bucket può contenere oggetti archiviati per tutte le classi di archiviazione.

- Si possono utilizzare le policy del ciclo di vita Amazon S3 per passare automaticamente gli oggetti tra le classi di archiviazione oppure rimuovere i dati senza modifiche all'applicazione. In generale, devi raggiungere un equilibrio tra efficienza delle risorse, latenza di accesso e affidabilità, quando consideri questi meccanismi di storage.

Risorse

Documenti correlati:

- Tipi di volume Amazon EBS
- Archivio dell'istanza Amazon EC2
- Intelligent Tiering di Amazon S3
- Caratteristiche di I/O Amazon EBS
- Utilizzo delle classi di archiviazione di Amazon S3
- Che cos'è Amazon S3 Glacier?

Video correlati:

- AWS re:Invent 2023 - Improve Amazon EBS efficiency and be more cost-efficient
- AWS re:Invent 2023 - Optimizing storage price and performance with Amazon S3
- AWS re:Invent 2023 - Building and optimizing a data lake on Amazon S3
- AWS re:Invent 2022 - Building modern data architectures on AWS
- AWS re:Invent 2022 - Modernize apps with purpose-built databases
- AWS re:Invent 2022 - Building data mesh architectures on AWS
- AWS re:Invent 2023 - Deep dive into Amazon Aurora and its innovations
- AWS re:Invent 2023 - Advanced data modeling with Amazon DynamoDB

Esempi correlati:

- Esempi di Amazon S3
- AWS Purpose Built Databases Workshop
- Databases for Developers
- AWS Modern Data Architecture Immersion Day
- Build a Data Mesh on AWS

SUS04-BP03 Utilizzo delle policy per gestire il ciclo di vita dei set di dati

Gestisci il ciclo di vita di tutti i tuoi dati e applica in automatico le cancellazioni per ridurre i requisiti totali di archiviazione del tuo carico di lavoro.

Anti-pattern comuni:

- Cancellazione manuale dei dati.
- Conservazione di tutti i dati del carico di lavoro.
- Mancato spostamento dei dati su livelli di archiviazione più efficienti dal punto di vista energetico in base ai requisiti di conservazione e accesso.

Vantaggi dell'adozione di questa best practice: l'utilizzo delle policy per il ciclo di vita dei dati garantisce un accesso e una conservazione efficienti dei dati in un carico di lavoro.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

i set di dati presentano solitamente requisiti di conservazione e accesso che cambiano durante il loro ciclo di vita. Ad esempio, l'applicazione potrebbe avere bisogno di accedere frequentemente ad alcuni set di dati per un periodo di tempo limitato. In seguito, questi set di dati vengono consultati di rado.

Per gestire in modo efficiente i set di dati durante il loro ciclo di vita, è necessario configurare le policy per il ciclo di vita, ovvero le regole che definiscono la gestione dei set di dati.

Con le regole di configurazione del ciclo di vita, è possibile indicare al servizio di archiviazione di trasferire un set di dati a livelli di archiviazione più efficienti dal punto di vista energetico, di archiviarlo o di eliminarlo.

Passaggi dell'implementazione

- Classifica i set di dati del carico di lavoro.
- Definisci le procedure di gestione per ogni classe di dati.
- Imposta policy automatizzate per il ciclo di vita per applicare le regole correlate. Ecco alcuni esempi di come impostare policy automatizzate per il ciclo di vita di diversi servizi di archiviazione di AWS:

Storage service	How to set automated lifecycle policies
Amazon S3	È possibile utilizzare Amazon S3 Lifecycle per gestire gli oggetti durante il loro ciclo di vita. Se gli schemi di accesso sono sconosciuti, mutevoli o imprevedibili, è possibile utilizzare Amazon S3 Intelligent-Tiering, che monitora gli schemi di accesso e sposta automaticamente gli oggetti che non hanno fatto registrare accessi a livelli di accesso più economici. Puoi sfruttare i parametri di Amazon S3 Storage Lens per identificare le opportunità di ottimizzazione e le lacune nella gestione del ciclo di vita.
Amazon Elastic Block Store	Puoi utilizzare Amazon Data Lifecycle Manager per automatizzare la creazione, la conservazione e l'eliminazione degli snapshot Amazon EBS e delle AMI basate su Amazon EBS.
Amazon Elastic File System	La gestione del ciclo di vita di Amazon EFS gestisce automaticamente l'archiviazione dei file per i tuoi file system.
Amazon Elastic Container Registry	Le policy per il ciclo di vita di Amazon ECR automatizzano la pulizia delle immagini dei container, facendo scadere le immagini in base all'età o al numero di copie.
AWS Elemental MediaStore	Si può usare una policy per il ciclo di vita di un oggetto che regola la durata di conservazione degli oggetti nel container MediaStore.

- Elimina i volumi inutilizzati, gli snapshot e i dati che hanno superato il periodo di conservazione. Sfrutta le funzionalità native del servizio, come il Time To Live di Amazon DynamoDB o la conservazione dei log di Amazon CloudWatch per programmare l'eliminazione.

- Aggrega e comprimi i dati quando possibile in base alle regole del ciclo di vita.

Risorse

Documenti correlati:

- Ottimizza le regole del ciclo di vita di Amazon S3 con l'analisi delle classi di archiviazione di Amazon S3
- Valutazione delle risorse con Regole di AWS Config

Video correlati:

- AWS re:Invent 2021 - Amazon S3 Lifecycle best practices to optimize your storage spend
- AWS re:Invent 2023 - Optimizing storage price and performance with Amazon S3
- Semplifica il ciclo di vita dei dati e ottimizza i costi di archiviazione con Amazon S3 Lifecycle
- Riduci i costi di archiviazione con Amazon S3 Storage Lens

SUS04-BP04 Utilizzo dell'elasticità e dell'automazione per espandere lo storage a blocchi o il file system

Usa l'elasticità e l'automazione per espandere lo storage a blocchi o il file system con l'aumento dei dati per ridurre l'archiviazione totale oggetto di provisioning.

Anti-pattern comuni:

- Acquisti uno storage a blocchi di grandi dimensioni o un file system per necessità future.
- Esegui un provisioning eccessivo delle operazioni di input e output al secondo (IOPS) del tuo file system.
- Non monitori l'utilizzo dei volumi di dati.

Vantaggi dell'adozione di questa best practice: ridurre il provisioning eccessivo per il sistema di archiviazione significa ridurre le risorse inattive e migliorare l'efficienza complessiva del tuo carico di lavoro.

Livello di rischio associato alla mancata adozione di questa best practice: medio

Guida all'implementazione

Crea storage a blocchi e file system con l'allocazione delle dimensioni, la velocità di trasmissione effettiva e la latenza adeguate al tuo carico di lavoro. Usa l'elasticità e l'automazione per espandere lo storage a blocchi o il file system con l'aumento dei dati per evitare un provisioning eccessivo per questi servizi di archiviazione.

Passaggi dell'implementazione

- Per i sistemi di storage di dimensioni fisse, come Amazon EBS, assicurati di monitorare la quantità di archiviazione utilizzata rispetto alle dimensioni complessive dell'archiviazione e di creare, se possibile, un'automazione per aumentarne le dimensioni quando si raggiunge una soglia.
- Utilizza volumi elastici e servizi di dati a blocchi gestiti per automatizzare l'allocazione di archivi aggiuntivi man mano che i dati persistenti aumentano. Come esempio puoi usare i Volumi elastici Amazon EBS per modificare le dimensioni dei volumi, il tipo di volume o adeguare le performance dei tuo volumi Amazon EBS.
- Scegli la classe di archiviazione corretta, le performance e la velocità di trasmissione effettiva per il tuo file system per rispondere alle esigenze della tua azienda, senza eccedere.
  - Performance Amazon EFS
  - Performance dei volumi Amazon EBS sulle istanze Linux
- Imposta i livelli target di utilizzo per i volumi di dati e ridimensiona i volumi al di fuori degli intervalli previsti.
- Dimensiona i volumi di sola lettura per adattarli ai dati.
- Migra i dati su archivi oggetti per evitare il provisioning di capacità eccessive da dimensioni di volumi fisse su archiviazioni a blocchi.
- Esamina regolarmente i volumi elastici e i file system per terminare i volumi inattivi e ridurre i volumi con un provisioning eccessivo per adattarli alla dimensione corrente dei dati.

Risorse

Documenti correlati:

- Extend the file system after resizing an EBS volume
- Modify a volume using Amazon EBS Elastic Volumes
- Documentazione Amazon FSx
- Che cos'è Amazon Elastic File System?

Video correlati:

- Approfondimento sui volumi elastici di Amazon EBS
- Strategie Amazon EBS e di ottimizzazione degli snapshot per performance migliori e risparmio sui costi
- Ottimizzare Amazon EFS per costi e performance, usando le best practice

SUS04-BP05 Eliminazione dei dati ridondanti o non necessari

Elimina i dati non necessari o ridondanti per ridurre al minimo le risorse di archiviazione necessarie per memorizzare i set di dati.

Anti-pattern comuni:

- Duplicazione dei dati che possono essere facilmente recuperati o ricreati.
- Backup di tutti i dati senza prenderne in considerazione la criticità.
- Cancellazione dei dati eseguita in modo irregolare, in occasione di eventi operativi o non eseguita affatto.
- Archiviazione dei dati in modo ridondante, indipendentemente dall'affidabilità del servizio di archiviazione.
- Attivazione del controllo delle versioni di Amazon S3 senza alcuna giustificazione aziendale.

Vantaggi dell'adozione di questa best practice: la rimozione dei dati non necessari riduce le dimensioni dello spazio di archiviazione necessario per il carico di lavoro e il relativo impatto ambientale.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Non memorizzare i dati che non ti servono. Automatizza l'eliminazione dei dati non necessari. Utilizza tecnologie di backup che deduplicano i dati a livello di file e blocco. Sfrutta le funzionalità native di replica e ridondanza dei dati dei servizi.

Passaggi dell'implementazione

- Valuta se è possibile evitare la memorizzazione dei dati utilizzando set di dati esistenti disponibili pubblicamente in AWS Data Exchange e Open Data su AWS.

- Utilizza meccanismi che possano deduplicare i dati a livello di blocco e oggetto. Ecco alcuni esempi di come deduplicare i dati su AWS:

Storage service	Deduplication mechanism
Amazon S3	Utilizza AWS Lake Formation FindMatches per individuare i record corrispondenti in un set di dati (compresi quelli senza identific atori), utilizzando il nuovo FindMatches ML Transform.
Amazon FSx	Usa la deduplicazione dei dati su Amazon FSx per Windows.
Snapshot di Amazon Elastic Block Store	Gli snapshot sono backup incrementali, il che significa che vengono salvati solo i blocchi sul dispositivo che sono stati modificati dopo lo snapshot più recente.

- Analizza l'accesso ai dati per identificare quelli non necessari. Automatizza le policy per il ciclo di vita. Sfrutta le caratteristiche native del servizio, come il Time To Live di Amazon DynamoDB, Amazon S3 Lifecycle o la conservazione dei log di Amazon CloudWatch per l'eliminazione.
- Utilizza le funzionalità di virtualizzazione dei dati di AWS per mantenere i dati sul loro sistema di origine ed evitare la loro duplicazione.
  - Cloud Native Data Virtualization on AWS
  - Optimize Data Pattern Using Amazon Redshift Data Sharing
- Utilizza una tecnologia di backup in grado di eseguire backup incrementali.
- Per raggiungere i tuoi obiettivi di persistenza, sfrutta l'affidabilità di Amazon S3 e la replica di Amazon EBS invece di tecnologie da gestire in autonomia (come i dischi RAID).
- Centralizza i log e traccia i dati, deduplica le voci di log identiche e stabilisci meccanismi per ottimizzarne la verbosità quando necessario.
- Popola in anticipo le cache solo quando è necessario.
- Definisci il monitoraggio e l'automazione della cache per ridimensionarla in base alle esigenze.
- Rimuovi le implementazioni e le risorse obsolete dagli archivi di oggetti e dalle cache edge durante la distribuzione di nuove versioni del carico di lavoro.

Risorse

Documenti correlati:

- Modifica la conservazione dei dati di log in CloudWatch Logs
- Deduplicazione dei dati su Amazon FSx per Windows File Server
- Funzionalità di Amazon FSx per ONTAP, compresa la deduplicazione dei dati
- Invalidazione dei file su Amazon CloudFront
- Utilizzo di AWS Backup per il backup e il ripristino dei file system di Amazon EFS
- Che cos'è Amazon CloudWatch Logs?
- Working with backups on Amazon RDS
- Integrate and deduplicate datasets using AWS Lake Formation

Video correlati:

- Amazon Redshift Data Sharing Use Cases

Esempi correlati:

- Come faccio ad analizzare i miei log di accesso al server Amazon S3 utilizzando Amazon Athena?

SUS04-BP06 Utilizzo di file system condivisi o archiviazione per accedere a dati comuni

Adotta file system condivisi o l'archiviazione per evitare duplicazioni di dati e abilitare un'infrastruttura più efficiente per il tuo carico di lavoro.

Anti-pattern comuni:

- Esegui il provisioning dell'archiviazione per ogni singolo client.
- Non scolleghi volumi di dati da client inattivi.
- Non fornisci l'accesso allo storage su piattaforme e sistemi.

Vantaggi dell'adozione di questa best practice: usare file system o archiviazioni condivisi consente di distribuire i dati a uno o più consumatori senza doverli copiare. Questo consente di ridurre le risorse di archiviazione necessarie per il carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Se hai più utenti o applicazioni che accedono agli stessi set di dati, usare una tecnologia di archiviazione condivisa è fondamentale per usare un'infrastruttura efficiente per il tuo carico di lavoro. La tecnologia di archiviazione condivisa offre una posizione centrale per archiviare e gestire set di dati ed evitare la loro duplicazione. Verifica anche la coerenza dei dati su sistemi diversi. Inoltre, la tecnologia di archiviazione condivisa consente un uso più efficiente della potenza di elaborazione, poiché più risorse di calcolo possono accedere ed elaborare i dati allo stesso momento in parallelo.

Acquisisci i dati dai servizi di archiviazione condivisa in base alle necessità e scollega i volumi non utilizzati per liberare le risorse.

Passaggi dell'implementazione

- Esegui la migrazione dei dati nell'archiviazione condivisa quando i dati hanno più consumer. Ecco alcuni esempi della tecnologia di archiviazione condivisa su AWS:

Storage option	When to use
Amazon EBS Multi-Attach	Amazon EBS Multi-Attach consente di collegare un volume singolo Provisioned IOPS SSD (io1 o io2) a più istanze che si trovano nella stessa zona di disponibilità.
Amazon EFS	See Quando scegliere Amazon EFS.
Amazon FSx	See Scegliere un Amazon FSx File System.
Amazon S3	Le applicazioni che non richiedono una struttura di file system e sono progettate per lavorare con lo storage degli oggetti possono usare Amazon S3 come soluzione di archiviaz ione degli oggetti a basso costo, durevole e altamente scalabile.

- Copia o acquisici i dati solo da file system condivisi in base alle necessità. Come esempio, puoi creare un file system Amazon FSx for Lustre supportato da Amazon S3 e caricare solo il sottoinsieme di dati richiesti per l'elaborazione dei processi su Amazon FSx.

- Elimina i dati nella modalità corretta per i tuoi modelli di utilizzo come definito in SUS04-BP03 Utilizzo delle policy per gestire il ciclo di vita dei set di dati.
- Distacca i volumi dai client che non li utilizzano attivamente.

Risorse

Documenti correlati:

- Collegare il file system a un bucket Amazon S3
- Usare Amazon EFS per AWS Lambda nelle applicazioni serverless
- Amazon EFS Intelligent-Tiering ottimizza i costi per carichi di lavoro con modelli di accesso mutevoli
- Uso di Amazon FSx con repository di dati on-premise

Video correlati:

- Storage cost optimization with Amazon EFS
- AWS re:Invent 2023 - What's new with AWS file storage
- AWS re:Invent 2023 - File storage for builders and data scientists on Amazon Elastic File System

SUS04-BP07 Riduzione al minimo dello spostamento di dati tra reti

Usa file system condivisi o lo storage a oggetti per accedere ai dati comuni e contenere le risorse di rete totali necessarie per supportare i trasferimenti dei dati per il carico di lavoro.

Anti-pattern comuni:

- Archivi tutti i dati nella stessa Regione AWS, indipendentemente dalla posizione degli utenti.
- Non ottimizzi la dimensione e il formato dei dati prima di trasferirli sulla rete.

Vantaggi dell'adozione di questa best practice: l'ottimizzazione del trasferimento dei dati sulla rete riduce la quantità di risorse di rete totali richieste per il carico di lavoro e diminuisce l'impatto ambientale.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Trasferire i dati all'interno dell'organizzazione significa disporre di risorse di elaborazione, rete e archiviazione. Usa tecniche per ridurre il movimento dei dati e migliorare l'efficienza generale del tuo carico di lavoro.

Passaggi dell'implementazione

- Considera la vicinanza ai dati o agli utenti come un fattore importante nella fase decisionale per la selezione di un'area geografica per il tuo carico di lavoro.
- Esegui la partizione dei servizi consumati a livello regionale in modo che i dati specifici della Regione siano archiviati nella Regione in cui sono usati.
- Usa formati di file efficienti (come Parquet oppure ORC) e comprimi i dati prima di spostarli sulla rete.
- Non trasferire dati inutilizzati. Alcuni esempi che possono aiutarti a evitare di spostare dati inutilizzati:
  - Riduci le risposte API solo ai dati pertinenti.
  - Aggrega i dati laddove richiesto (le informazioni a livello di record non sono necessarie).
  - Consulta Well-Architected Lab - Optimize Data Pattern Using Amazon Redshift Data Sharing.
  - Consulta Cross-account data sharing in AWS Lake Formation.
- Utilizza servizi in grado di supportarti nell'esecuzione del codice in posizioni più vicine agli utenti del carico di lavoro.

Service	When to use
Lambda@Edge	Usa per operazioni a uso intensivo di risorse di calcolo eseguite quando gli oggetti non si trovano nella cache.
Funzioni CloudFront	Usa per casi d'uso semplici, ad esempio manipolazioni di risposte o richieste HTTP(s) che possono essere avviate da funzioni di breve durata.

Service	When to use
AWS IoT Greengrass	Eseguire la memorizzazione nella cache di risorse di calcolo, messaggistica e dati per i dispositivi connessi.

Risorse

Documenti correlati:

- Ottimizzazione dell'infrastruttura AWS per la sostenibilità, parte III: reti
- Infrastruttura globale AWS
- Funzionalità principali di Amazon CloudFront, tra cui CloudFront Global Edge Network
- Compressione delle richieste HTTP in Amazon OpenSearch Service
- Compressione intermedia dei dati con Amazon EMR
- Caricamento di file di dati compressi da Amazon S3 a Amazon Redshift
- Distribuzione dei file compressi con Amazon CloudFront

Video correlati:

- Demystifying data transfer on AWS

Esempi correlati:

- Architecting for sustainability - Minimize data movement across networks

SUS04-BP08 Backup dei dati solo quando sono difficili da ricreare

Evita il back-up di dati senza valore aziendale per ridurre i requisiti delle risorse di archiviazione per il tuo carico di lavoro.

Anti-pattern comuni:

- Non hai una strategia di back-up per i tuoi dati.
- Esegui il back-up di dati che possono essere facilmente ricreati.

Vantaggi dell'adozione di questa best practice: se si evita il back-up di dati non critici si riduce la quantità di risorse di archiviazione richiesta per il carico di lavoro e si diminuisce l'impatto ambientale.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Evitando il back-up di dati non necessari si possono ridurre i costi e le risorse di archiviazione utilizzate dal carico di lavoro. Esegui il backup solo dei dati che hanno un valore aziendale o sono considerati necessari per soddisfare i requisiti di conformità. Esamina le policy di backup ed escludi l'archiviazione temporanea che non offre valore in uno scenario di ripristino.

Passaggi dell'implementazione

- Implementazione di una policy di classificazione dei dati come definito in SUS04-BP01 Implementazione di una policy di classificazione dei dati.
- Usa le criticità della classificazione dei tuoi dati e la strategia di backup della progettazione basate su Obiettivo del tempo di ripristino (RTO) e Obiettivo del punto di ripristino (RPO). Evita il back-up di dati non critici.
  - Escludi i dati che possono essere facilmente ricreati.
  - Escludi dati temporanei dai backup.
  - Escludi copie locali dei dati, a meno che il tempo necessario per ripristinare tali dati da una posizione comune superi gli accordi sul livello di servizio (SLA).
- Usa una soluzione automatizzata o un servizio gestito per eseguire il back-up di dati aziendali strategici.
  - AWS Backup è un servizio completamente gestito che semplifica la centralizzazione e l'automazione della protezione dei dati tra i servizi AWS, nel cloud e on-premise. Per linee guida pratiche su come creare back-up automatizzati con AWS Backup, consulta Well-Architected Labs: Test di backup e ripristino dei dati.
  - Automatizza i back-up e ottimizza i costi di back-up per Amazon EFS con AWS Backup.

Risorse

Best practice correlate:

- REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o riproduzione dei dati dalle origini
- REL09-BP03 Esecuzione del backup dei dati in automatico

- REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino

Documenti correlati:

- Utilizzo di AWS Backup per il backup e il ripristino dei file system di Amazon EFS
- Snapshot di Amazon EBS
- Lavorare con i backup su Amazon Relational Database Service
- Partner APN: partner per il backup
- Marketplace AWS: prodotti che possono essere utilizzati per il backup
- Backup di Amazon EFS
- Backup Amazon FSx per Windows File Server
- Backup e ripristino per Amazon ElastiCache for Redis

Video correlati:

- AWS re:Invent 2023 - Backup and disaster recovery strategies for increased resilience
- AWS re:Invent 2023 - What's new with AWS Backup
- AWS re:Invent 2021 - Backup, disaster recovery, and ransomware protection with AWS

Esempi correlati:

- Well-Architected Lab - Backup data

Hardware e servizi

Domanda

- SUS 5 Come si selezionano e usano hardware e servizi cloud nell'architettura per supportare gli obiettivi di sostenibilità?

SUS 5 Come si selezionano e usano hardware e servizi cloud nell'architettura per supportare gli obiettivi di sostenibilità?

Cerca opportunità per ridurre l'impatto dei carichi di lavoro in termini di sostenibilità apportando modifiche alle tue prassi di gestione hardware. Riduci al minimo la quantità di hardware necessaria

per il provisioning e l'implementazione e scegli l'hardware e i servizi più efficienti per il singolo carico di lavoro.

Best practice

- SUS05-BP01 Utilizzo della quantità minima di hardware per soddisfare le esigenze aziendali
- SUS05-BP02 Utilizzo di tipi di istanze con il minimo impatto
- SUS05-BP03 Utilizzo dei servizi gestiti
- SUS05-BP04 Ottimizzazione dell'uso degli acceleratori di calcolo basati su hardware

SUS05-BP01 Utilizzo della quantità minima di hardware per soddisfare le esigenze aziendali

Usa la quantità minima di hardware per il tuo carico di lavoro per soddisfare in modo efficiente le tue esigenze aziendali.

Anti-pattern comuni:

- Non monitori l'utilizzo delle risorse.
- Nella tua architettura sono presenti risorse con un basso livello di utilizzo.
- Non analizzi l'uso di hardware statico per stabilire se deve essere ridimensionato.
- Non imposti obiettivi di utilizzo dell'hardware per la tua infrastruttura di elaborazione in base a KPI aziendali.

Vantaggi dell'adozione di questa best practice: il corretto dimensionamento delle risorse cloud consente di ridurre l'impatto ambientale di un carico di lavoro, risparmiare denaro ed essere in linea con i benchmark di performance.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Seleziona con precisione la quantità di hardware richiesta dal tuo carico di lavoro per migliorare l'efficienza generale. Il Cloud AWS offre la flessibilità necessaria per espandere o ridurre il numero di risorse in modo dinamico attraverso una serie di meccanismi, come AWS Auto Scaling e soddisfare i cambiamenti della domanda. AWS offre anche API e SDK che consentono alle risorse di essere modificate con il minimo sforzo. usa queste funzionalità per apportare modifiche frequenti alle implementazioni dei carichi di lavoro. Usa inoltre le linee guida sul dimensionamento corretto degli strumenti AWS per gestire le risorse cloud in modo efficiente e soddisfare le esigenze aziendali.

Passaggi dell'implementazione

- Scegli il tipo di istanza: scegli il tipo di istanza giusto per soddisfare al meglio le tue esigenze. Per informazioni su come scegliere le istanze Amazon Elastic Compute Cloud e utilizzare meccanismi quali la selezione delle istanze basata sugli attributi, consulta le seguenti risorse:
  - Come faccio a scegliere il tipo di istanza Amazon EC2 appropriato per il mio carico di lavoro?
  - Selezione del tipo di istanza basata sugli attributi per il parco istanze Amazon EC2.
  - Create an Auto Scaling group using attribute-based instance type selection.
- Scala: utilizza piccoli incrementi per scalare i carichi di lavoro variabili.
- Utilizza più opzioni di acquisto di calcolo: bilancia flessibilità, scalabilità e risparmio sui costi delle istanze con più opzioni di acquisto di calcolo.
  - Le istanze on-demand Amazon EC2 sono più adatte a carichi di lavoro nuovi, stateful e con picchi che non possono essere flessibili dal punto di vista dei tipi di istanza, della posizione o dal punto di vista temporale.
  - Le istanze spot Amazon EC2 sono un ottimo modo per fornire altre opzioni alle applicazioni flessibili e tolleranti ai guasti.
  - Sfrutta i Savings Plans per il calcolo per carichi di lavoro stazionari che consentono soluzioni flessibili se le tue esigenze (come AZ, area geografica, famiglie di istanze o tipi di istanze) cambiano.
- Usa la diversità di istanze e zone di disponibilità: massimizza la disponibilità delle applicazioni e sfrutta la capacità in eccesso diversificando le istanze e le zone di disponibilità.
- Dimensiona correttamente le istanze: utilizza i suggerimenti per il dimensionamento corretto degli strumenti AWS per regolare il carico di lavoro. Per ulteriori informazioni, consulta Optimizing your cost with Rightsizing Recommendations e Right Sizing: Provisioning Instances to Match Workloads
  - Puoi utilizzare i suggerimenti per il dimensionamento corretto in AWS Cost Explorer o AWS Compute Optimizer per identificare le corrette opportunità di dimensionamento.
- Negozia gli accordi sul livello di servizio (service-level agreement, SLA): negozia gli SLA che consentono di ridurre temporaneamente la capacità mentre l'automazione implementa le risorse sostitutive.

Risorse

Documenti correlati:

- Ottimizzazione dell'infrastruttura AWS per la sostenibilità, Parte I: elaborazione

- Selezione del tipo di istanza basata sugli attributi per Auto Scaling per il parco istanze Amazon EC2
- Documentazione di AWS Compute Optimizer
- Uso di Lambda: ottimizzazione delle performance
- Documentazione sulla scalabilità automatica

Video correlati:

- AWS re:Invent 2023 - What's new with Amazon EC2
- AWS re:Invent 2023 - Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies
- AWS re:Invent 2022 - Optimizing Amazon Elastic Kubernetes Service for performance and cost on AWS
- AWS re:Invent 2023 - Sustainable compute: reducing costs and carbon emissions with AWS

SUS05-BP02 Utilizzo di tipi di istanze con il minimo impatto

Esegui un monitoraggio costante e usa nuovi tipi di istanza per sfruttare le migliorie in termini di efficienza energetica.

Anti-pattern comuni:

- Utilizzi una sola famiglia di istanze.
- Utilizzi solo istanze x86.
- Specifichi un tipo di istanza nella configurazione Amazon EC2 Auto Scaling.
- Utilizzi istanze AWS in un modo per il quale non sono state progettate, ad esempio utilizzi istanze ottimizzate per il calcolo per un carico di lavoro a uso intensivo della memoria.
- Non valuti regolarmente l'uso di nuovi tipi di istanza.
- Non segui i consigli ricevuti dagli strumenti di dimensionamento AWS, ad esempio AWS Compute Optimizer.

Vantaggi dell'adozione di questa best practice: l'uso di istanze energeticamente efficienti e di dimensioni corrette consente di ridurre in modo considerevole l'impatto ambientale e i costi del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

L'uso di istanze efficienti nel carico di lavoro cloud è fondamentale per ridurre l'utilizzo delle risorse e i costi. Monitora costantemente il rilascio di nuovi tipi di istanza e sfrutta le migliorie in tema di efficienza energetica, inclusi i tipi di istanza progettati per supportare carichi di lavoro specifici, come la formazione del machine learning, le inferenze e la transcodifica dei video.

Passaggi dell'implementazione

- Scopri e approfondisci i tipi di istanze: individua i tipi di istanze che possono ridurre l'impatto ambientale del tuo carico di lavoro.
  - Iscriviti alle Novità di AWS per rimanere aggiornato sulle più recenti tecnologie e istanze AWS.
  - Approfondisci i vari tipi di istanza AWS.
  - Scopri di più sulle istanze basate su AWS Graviton che offrono le performance migliori performance per watt di energia utilizzata in Amazon EC2 guardando re:Invent 2020 - Approfondimento sulle istanze Amazon EC2 alimentate dal processore AWS Graviton2 e Approfondimento sulle istanze AWS Graviton3 e Amazon EC2 C7g.
- Usa i tipi di istanza che comportano il minor impatto: pianifica la transizione del carico di lavoro ai tipi di istanza con il minor impatto.
  - Definisci un processo per valutare nuove funzionalità o istanze per il carico di lavoro. Sfrutta l'agilità del cloud per testare in modo semplice e rapido in che modo i nuovi tipi di istanza possono migliorare la sostenibilità ambientale del carico di lavoro. Utilizza metriche proxy per misurare la quantità di risorse necessarie per completare un'unità di lavoro.
  - Se possibile, modifica il carico di lavoro in modo che funzioni con diversi numeri di CPU e quantità di memoria diverse per massimizzare la scelta del tipo di istanza.
  - Valuta l'ipotesi di trasferire il carico di lavoro in istanze basate su Graviton per migliorare l'efficienza delle prestazioni del carico di lavoro. Per ulteriori informazioni sullo spostamento dei carichi di lavoro su AWS Graviton, consulta AWS Graviton Fast Start e Considerations when transitioning workloads to AWS Graviton-based Amazon Elastic Compute Cloud instances.
  - Valuta l'ipotesi di selezionare l'opzione AWS Graviton quando utilizzi i servizi gestiti AWS.
  - Esegui la migrazione del carico di lavoro nelle regioni che offrono istanze con il minor impatto in termini di sostenibilità e che contemporaneamente soddisfano i requisiti aziendali.
  - Per i carichi di lavoro di machine learning, sfrutta l'hardware specifico per il tuo carico di lavoro, come ad esempio AWS Trainium, AWS Inferentia e Amazon EC2 DL1. Le istanze AWS Inferentia come Inf2 offrono fino al 50% in più di prestazioni per watt rispetto alle istanze Amazon EC2 paragonabili.

  - Usa Amazon SageMaker Inference Recommender per dimensionare correttamente l'endpoint dell'inferenza ML.
  - Per carichi di lavoro con picchi (carichi di lavoro con requisiti non frequenti di capacità aggiuntiva), utilizza istanze a prestazioni espandibili.
  - Usa le Istanze spot Amazon EC2 per carichi di lavoro stateless e con tolleranza ai guasti per aumentare l'utilizzo complessivo del cloud e ridurre l'impatto di sostenibilità delle risorse inutilizzate.
- Esegui e ottimizza: esegui e ottimizza l'istanza del carico di lavoro.
  - Per carichi di lavoro effimeri valuta le metriche Amazon CloudWatch dell'istanza, come `CPUUtilization` per verificare se l'istanza è inattiva o sottoutilizzata.
  - Per i carichi di lavoro stabili, esegui controlli con gli strumenti di ridimensionamento AWS, come AWS Compute Optimizer a intervalli regolari per individuare eventuali opportunità di ottimizzazione e ridimensionamento corretto delle istanze.
    - Well-Architected Lab: Raccomandazioni per il dimensionamento appropriato
    - Well-Architected Lab: Ridimensionamento con Compute Optimizer
    - Well-Architected Lab: ottimizzazione dei modelli hardware e conformità agli indicatori KPI di sostenibilità

Risorse

Documenti correlati:

- Ottimizzazione dell'infrastruttura AWS per la sostenibilità, Parte I: elaborazione
- AWS Graviton
- Amazon EC2 DL1
- Parchi istanza di prenotazione della capacità Amazon EC2
- Serie di istanze spot Amazon EC2
- Funzioni: configurazione della funzione Lambda
- Attribute-based instance type selection for Amazon EC2 Fleet
- Building Sustainable, Efficient, and Cost-Optimized Applications on AWS
- How the Contino Sustainability Dashboard Helps Customers Optimize Their Carbon Footprint

Video correlati:

- AWS re:Invent 2023 - AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 - New Amazon Elastic Compute Cloud generative AI capabilities in AWS Management Console
- AWS re:Invent 2023 - What's new with Amazon Elastic Compute Cloud
- AWS re:Invent 2023 - Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies
- AWS re:Invent 2021 - Deep dive into AWS Graviton3 and Amazon EC2 C7g instances
- AWS re:Invent 2022 - Build a cost-, energy-, and resource-efficient compute environment

Esempi correlati:

- Solution: Guidance for Optimizing Deep Learning Workloads for Sustainability on AWS
- Migrating Amazon Relational Database Service Databases to Graviton

SUS05-BP03 Utilizzo dei servizi gestiti

Usa i servizi gestiti per operare in modo più efficiente nel cloud.

Anti-pattern comuni:

- Usi istanze Amazon EC2 in modo ridotto per eseguire le tue applicazioni.
- Il tuo team interno gestisce solo il carico di lavoro, senza tempo per focalizzarsi sull'innovazione o sulle semplificazioni.
- Implementi e mantieni tecnologie per attività che possono essere eseguite in modo più efficiente sui servizi gestiti.

Vantaggi dell'adozione di questa best practice:

- L'uso dei servizi gestiti sposta la responsabilità su AWS, che ha visibilità su milioni di clienti, i quali possono contribuire alla promozione di nuove innovazioni ed efficienze.
- Il servizio gestito distribuisce l'impatto ambientale del servizio su molti utenti a causa dei piani di controllo multi-tenet.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I servizi gestiti consentono di affidare ad AWS la responsabilità di mantenere un utilizzo alto e un'ottimizzazione della sostenibilità dell'hardware implementato. I servizi gestiti eliminano anche l'onere operativo e amministrativo legato alla manutenzione di un servizio, consentendo al tuo team di avere più tempo e di concentrarsi sull'innovazione.

Esamina il carico di lavoro per identificare i componenti che possono essere sostituiti dai servizi gestiti AWS. Ad esempio, Amazon RDS, Amazon Redshift e Amazon ElastiCache offrono un servizio di database gestito. Amazon Athena, Amazon EMR e Amazon OpenSearch Service offrono un servizio di analisi gestito.

Passaggi dell'implementazione

1. Esegui l'inventario del carico di lavoro: esegui l'inventario di servizi e componenti del tuo carico di lavoro.
2. Identifica i candidati: valuta e identifica i componenti che possono essere sostituiti dai servizi gestiti. Ecco alcuni esempi in cui potresti prendere in considerazione l'uso di un servizio gestito:

Task	What to use on AWS
Ospitare un database	Usa istanze gestite Amazon Relational Database Service (Amazon RDS) invece di mantenere le tue istanze Amazon RDS su Amazon Elastic Compute Cloud (Amazon EC2).
Ospitare il carico di lavoro di un container	Usa AWS Fargate, invece di implementare un'infrastruttura di container proprietaria.
Ospitare applicazioni Web	Usa l'Hosting AWS Amplify come CI/CD completamente gestito e servizio di hosting per siti Web statici e app Web con rendering lato server.

3. Crea un piano di migrazione: identifica le dipendenze e crea un piano di migrazione. Aggiorna runbook e playbook.

- AWS Application Discovery Service raccoglie e illustra automaticamente informazioni dettagliate sulle dipendenze delle applicazioni e sul loro utilizzo per aiutarti a prendere decisioni più informate durante la pianificazione della migrazione.

4. Esegui i test: esegui i test prima di migrare al servizio gestito.
5. Sostituisci i servizi self-hosted: usa il piano di migrazione per sostituire i servizi self-hosted con servizi gestiti.
6. Monitora e modifica: monitora costantemente il servizio al termine della migrazione per apportare le modifiche richieste e ottimizzare il servizio.

Risorse

Documenti correlati:

- Prodotti Cloud AWS
- Calcolatore del costo totale di proprietà (TCO) di AWS
- Amazon DocumentDB
- Amazon Elastic Kubernetes Service (EKS)
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)

Video correlati:

- AWS re:Invent 2021 - Cloud operations at scale with AWS Managed Services
- AWS re:Invent 2023 - Best practices for operating on AWS

SUS05-BP04 Ottimizzazione dell'uso degli acceleratori di calcolo basati su hardware

Ottimizza l'uso delle istanze a calcolo accelerato per ridurre i requisiti dell'infrastruttura fisica del carico di lavoro.

Anti-pattern comuni:

- Utilizzo delle GPU non monitorato.
- Utilizzo di un'istanza generica per il carico di lavoro quando un'istanza appositamente sviluppata potrebbe offrire prestazioni più elevate, costi inferiori e migliori prestazioni per watt.
- Utilizzo di acceleratori di calcolo basati su hardware per attività in cui sono più efficienti le alternative basate su CPU.

Vantaggi dell'adozione di questa best practice: ottimizzando l'uso degli acceleratori basati su hardware, è possibile ridurre le richieste di infrastruttura fisica del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Se si necessita di un'elevata capacità di elaborazione, si può trarre vantaggio dall'uso di istanze a calcolo accelerato, che forniscono l'accesso ad acceleratori di calcolo basati su hardware, come le unità di elaborazione grafica (GPU) e i field programmable gate array (FPGA). Questi acceleratori hardware eseguono alcune funzioni, come l'elaborazione grafica o la rilevazione della corrispondenza dei modelli di dati, in modo più efficiente rispetto alle alternative basate su CPU. Molti carichi di lavoro accelerati, come il rendering grafico, la transcodifica e il machine learning, sono altamente variabili in termini di utilizzo di risorse. Mantieni in esecuzione questo tipo di hardware solo per il tempo necessario e disattivalo automaticamente quando non serve per ridurre la quantità di risorse utilizzate.

Passaggi dell'implementazione

- Identifica quali istanze a calcolo accelerato possono soddisfare i tuoi requisiti.
- Per i carichi di lavoro di machine learning, sfrutta l'hardware specifico per il tuo carico di lavoro, come ad esempio AWS Trainium, AWS Inferentia e Amazon EC2 DL1. Le istanze AWS Inferentia come Inf2 offrono fino al 50% in più di prestazioni per watt rispetto alle istanze Amazon EC2 paragonabili.
- Raccogli i parametri di utilizzo delle istanze a calcolo accelerato. Ad esempio, puoi utilizzare l'agente CloudWatch per raccogliere parametri come `utilization_gpu` e `utilization_memory` per le GPU come mostrato in Raccolta dei parametri delle GPU NVIDIA con Amazon CloudWatch.
- Ottimizza il codice, il funzionamento della rete e le impostazioni degli acceleratori hardware per garantire il pieno utilizzo dell'hardware sottostante.
  - Ottimizza l impostazioni delle GPU
  - Monitoraggio e ottimizzazione delle GPU nell'AMI per il Deep Learning
  - Ottimizzazione dell'I/O per la messa a punto delle prestazioni delle GPU dedicate all'addestramento del deep learning in Amazon SageMaker
- Utilizzate le librerie e i driver per GPU più recenti e performanti.
- Utilizza l'automazione per rilasciare le istanze GPU non in uso.

Risorse

Documenti correlati:

- Calcolo accelerato
- Progettiamo! Sviluppo di architetture con chip e acceleratori personalizzati
- Come faccio a scegliere il tipo di istanza Amazon EC2 appropriato per il mio carico di lavoro?
- Istanze Amazon EC2 VT1
- Scegli il miglior acceleratore IA e compilatore del modello per l'inferenza nella visione artificiale con Amazon SageMaker

Video correlati:

- AWS re:Invent 2021 - How to select Amazon EC2 GPU instances for deep learning
- AWS Online Tech Talks - Deploying Cost-Effective Deep Learning Inference
- AWS re:Invent 2023 - Cutting-edge AI with AWS and NVIDIA
- AWS re:Invent 2022 - [NEW LAUNCH!] Introducing AWS Inferentia2-based Amazon EC2 Inf2 instances
- AWS re:Invent 2022 - Accelerate deep learning and innovate faster with AWS Trainium
- AWS re:Invent 2022 - Deep learning on AWS with NVIDIA: From training to deployment

Processo e cultura

Domanda

- SUS 6 In che modo i processi organizzativi possono supportare gli obiettivi di sostenibilità?

SUS 6 In che modo i processi organizzativi possono supportare gli obiettivi di sostenibilità?

Cerca opportunità per ridurre l'impatto di sostenibilità apportando modifiche alle tue prassi di sviluppo, test e implementazione.

Best practice

- SUS06-BP01 Adozione di metodi che consentano di introdurre rapidamente migliorie in tema di sostenibilità

- SUS06-BP02 Aggiornamento del carico di lavoro
- SUS06-BP03 Aumento dell'utilizzo degli ambienti di costruzione
- SUS06-BP04 Utilizzo di device farm gestite per i test

SUS06-BP01 Adozione di metodi che consentano di introdurre rapidamente migliorie in tema di sostenibilità

Adotta metodi e processi per convalidare migliorie potenziali, ridurre i costi legati ai test e offrire piccole migliorie.

Anti-pattern comuni:

- Analizzare l'applicazione rispetto alla sostenibilità è un'attività che viene eseguita solo una volta, all'inizio di un progetto.
- Il tuo carico di lavoro non è aggiornato, poiché il processo di rilascio è troppo complesso per introdurre modifiche minori per l'efficienza delle risorse.
- Non hai meccanismi per migliorare il tuo carico di lavoro in termini di sostenibilità.

Vantaggi dell'adozione di questa best practice: definendo un processo per avviare e monitorare le migliorie in termini di sostenibilità, potrai adottare continuamente nuove funzionalità, eliminare i problemi e migliorare l'efficienza del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Testa e convalida potenziali miglioramenti all'impatto sulla sostenibilità prima di implementarli in produzione. Tieni in considerazione il costo dei test quando calcoli il potenziale vantaggio futuro di un miglioramento. Sviluppa metodi di test a basso costo per consentire la distribuzione di piccoli miglioramenti.

Passaggi dell'implementazione

- Comprendi e comunica gli obiettivi di sostenibilità organizzativa: comprendi gli obiettivi di sostenibilità organizzativa, come la riduzione delle emissioni di carbonio o la gestione delle risorse idriche. Traduci questi obiettivi in requisiti di sostenibilità per i carichi di lavoro del cloud. Comunica questi requisiti alle principali parti interessate.

- Aggiungi i requisiti di sostenibilità al backlog: aggiungi i requisiti per il miglioramento della sostenibilità al backlog di sviluppo.
- Itera e migliora: usa un processo di miglioramento iterativo che ti consente di identificare, valutare, dare la priorità, testare e implementare questi miglioramenti.
- Esegui il test utilizzando il prodotto minimo funzionante (MVP): sviluppa e testa i potenziali miglioramenti utilizzando i componenti rappresentativi del prodotto minimo funzionante per ridurre i costi e l'impatto ambientale dei test.
- Semplifica il processo: migliora e semplifica costantemente i processi di sviluppo. Ad esempio, automatizza il processo di distribuzione del software con pipeline di distribuzione e integrazione continue (CI/CD) per testare e implementare migliorie potenziali per ridurre il livello di impegno e gli errori causati da processi manuali.
- Gestisci formazione e sensibilizzazione: gestisci i programmi di formazione per i membri del team per istruirli sulla sostenibilità e sull'impatto delle attività rispetto agli obiettivi di sostenibilità organizzativa.
- Valuta e modifica: valuta continuamente l'impatto dei miglioramenti e apporta le modifiche necessarie.

Risorse

Documenti correlati:

- AWS offre soluzioni di sostenibilità
- Procedure di sviluppo agile e scalabile basate su AWS CodeCommit

Video correlati:

- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2022 - Architecting sustainably and reducing your AWS carbon footprint
- AWS re:Invent 2022 - Sustainability in AWS global infrastructure
- AWS re:Invent 2023 - What’s new with AWS observability and operations

Esempi correlati:

- Well-Architected Lab - Trasformare i report su costi e utilizzo in report sull'efficienza

SUS06-BP02 Aggiornamento del carico di lavoro

Aggiorna il tuo carico di lavoro per adottare funzionalità efficienti, eliminare le problematiche e migliorare l'efficienza generale del tuo carico di lavoro.

Anti-pattern comuni:

- Ritieni che l'architettura corrente diventi statica e non venga aggiornata nel corso del tempo.
- Non disponi di sistemi né esegui regolarmente una valutazione per la compatibilità di software e pacchetti aggiornati con il carico di lavoro.

Vantaggi dell'adozione di questa best practice: la definizione di un processo per garantire il costante aggiornamento del carico di lavoro ti consentirà di adottare nuove caratteristiche e funzionalità, risolvere i problemi e migliorare l'efficienza del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Sistemi operativi, runtime, middleware, librerie e applicazioni aggiornati possono incidere sull'efficienza dei carichi di lavoro e facilitano l'adozione delle tecnologie più efficienti. Il software aggiornato potrebbe anche includere funzionalità per misurare in modo più accurato l'impatto in termini di sostenibilità del carico di lavoro, poiché i fornitori offrono caratteristiche per raggiungere i propri obiettivi di sostenibilità. Adotta una cadenza regolare per aggiornare il tuo carico di lavoro con le ultime funzionalità e i rilasci più recenti.

Passaggi dell'implementazione

- Definisci un processo: usa un processo e una pianificazione per valutare nuove funzionalità o istanze per il carico di lavoro. Sfrutta l'agilità del cloud per testare in modo semplice e rapido il modo in cui le nuove funzionalità possono migliorare il carico di lavoro nei seguenti ambiti:
  - Riduzione dell'impatto a livello di sostenibilità.
  - Raggiungimento di maggiore efficienza in termini di prestazioni.
  - Eliminazione delle barriere finalizzata a un miglioramento pianificato.
  - Miglioramento della capacità di misurare e gestire l'impatto a livello di sostenibilità.
- Esegui l'inventario: redigi l'inventario del software e dell'architettura del carico di lavoro e identifica i componenti che richiedono un aggiornamento.

  - Puoi usare AWS Systems Manager Inventory per raccogliere i metadati relativi a sistema operativo (SO), applicazioni e istanze dalle istanze Amazon EC2 per avere una panoramica immediata su quali istanze stanno eseguendo il software e le configurazioni richieste dalle policy software e quali istanze devono essere aggiornate.
- Apprendi la procedura di aggiornamento: scopri come aggiornare i componenti del carico di lavoro.

Workload component	How to update
Immagini della macchina	Usa EC2 Image Builder per gestire gli aggiornamenti alle Amazon Machine Images (AMI) per le immagini Linux o Windows Server.
Immagini del container	Usa Amazon Elastic Container Registry (Amazon ECR) con la pipeline esistente per gestire le immagini Amazon Elastic Container Service (Amazon ECS).
AWS Lambda	AWS Lambda include funzionalità di gestione delle versioni.

- Utilizza l'automazione: usa l'automazione degli aggiornamenti per ridurre il livello di impegno per implementare le nuove funzionalità e limitare gli errori causati dai processi manuali.
  - Puoi usare CI/CD per aggiornare automaticamente le AMI, le immagini di container e altri artefatti relativi alla tua applicazione cloud.
  - Puoi usare strumenti come AWS Systems Manager Patch Manager per automatizzare il processo degli aggiornamenti di sistema e pianificare le attività tramite Finestre di manutenzione AWS Systems Manager.

Risorse

Documenti correlati:

- Centro di progettazione AWS
- Le novità di AWS
- Strumenti per sviluppatori in AWS

Video correlati:

- AWS re:Invent 2022 - Optimize your AWS workloads with best-practice guidance
- All Things Patch: AWS Systems Manager

Esempi correlati:

- Well-Architected Labs: Inventario e gestione delle patch
- Laboratorio: AWS Systems Manager

SUS06-BP03 Aumento dell'utilizzo degli ambienti di costruzione

Aumenta l'uso delle risorse per sviluppare, testare e creare i tuoi carichi di lavoro.

Anti-pattern comuni:

- Esegui il provisioning manuale o interrompi i tuoi ambienti di sviluppo.
- Fai in modo che i tuoi ambienti di sviluppo siano in esecuzione indipendentemente dalle attività di test, creazione o rilascio (ad esempio, eseguire un ambiente al di fuori dell'orario di lavoro dei membri del tuo team di sviluppo).
- Esegui un provisioning eccessivo delle tue risorse per gli ambienti di creazione.

Vantaggi derivanti dall'adozione di questa best practice: aumentando l'uso degli ambienti di sviluppo, puoi migliorare l'efficienza complessiva del tuo carico di lavoro cloud, allocando al contempo le risorse di cui gli sviluppatori hanno bisogno per creare, testare e sviluppare in modo efficiente.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Utilizza l'automazione e l'infrastruttura come codice per rendere operativi gli ambienti di produzione quando necessario e dismetterli quando non vengono utilizzati. Un modello comune consiste nel pianificare periodi di disponibilità che coincidano con l'orario di lavoro dei membri del team incaricati dello sviluppo. Gli ambienti di test devono essere molto simili alla configurazione di produzione. Tuttavia, cerca la possibilità di utilizzare tipi di istanze con capacità di espansione, istanze Spot Amazon EC2, servizi di database con dimensionamento automatico, container e tecnologie serverless per allineare la capacità di sviluppo e test all'uso. Limita i volumi di dati per soddisfare solo

i requisiti di test. Se usi i dati di produzione per i test, rifletti sulla possibilità di condividere i dati di produzione invece di spostarli.

Passaggi dell'implementazione

- Usa l'infrastructure as code: usa l'infrastructure as code per eseguire il provisioning dei tuoi ambienti di compilazione.
- Usa l'automazione: utilizza l'automazione per gestire il ciclo di vita degli ambienti di sviluppo e test e massimizzare l'efficienza delle tue risorse di compilazione.
- Ottimizza l'utilizzo: usa le strategie per ottimizzare l'utilizzo degli ambienti di sviluppo e test.
  - Utilizza ambienti rappresentativi minimi realizzabili per lo sviluppo e il test di potenziali miglioramenti.
  - Utilizza tecnologie serverless, se possibile.
  - Utilizza istanze on-demand per integrare i dispositivi per gli sviluppatori.
  - Utilizza i tipi di istanze con capacità di espansione, istanze Spot e altre tecnologie per allineare la capacità di compilazione all'uso.
  - Adotta servizi cloud nativi per un accesso sicuro alle shell delle istanze invece di implementare parchi istanze di host bastion.
  - Dimensiona automaticamente le tue risorse di sviluppo in base alle tue attività.

Risorse

Documenti correlati:

- AWS Systems Manager Session Manager
- Istanze espandibili di prestazioni Amazon EC2
- Che cos'è AWS CloudFormation?
- Che cos'è AWS CodeBuild?
- Instance Scheduler su AWS

Video correlati:

- AWS re:Invent 2023 - Continuous integration and delivery for AWS

SUS06-BP04 Utilizzo di device farm gestite per i test

Usa device farm gestite per testare in maniera efficiente una nuova funzionalità su un set rappresentativo di hardware.

Anti-pattern comuni:

- Testa e distribuisci manualmente la tua applicazione su singoli dispositivi fisici.
- Non usare il servizio di test delle app per testare e interagire con le tue app (ad esempio, Android, iOS e app Web) su dispositivi fisici reali.

Vantaggi dell'adozione di questa best practice: usare le device farm gestite per testare applicazioni abilitate al cloud offre una serie di vantaggi:

- Offrono funzionalità più efficienti per testare le applicazioni su un'ampia gamma di dispositivi.
- Eliminano la necessità di un'infrastruttura in-house per i test.
- Offrono diverse tipologie di dispositivi, tra cui hardware di generazioni precedenti e meno diffuso, eliminando così la necessità di aggiornamenti non necessari dei dispositivi.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

L'uso di device farm gestite può aiutarti a semplificare il processo di test per le nuove funzionalità su un gruppo rappresentativo di hardware. Le device farm gestite offrono diversi tipi di dispositivi, inclusi hardware meno diffusi e di generazioni precedenti, ed evitano l'impatto sulla sostenibilità dei clienti dovuti ad aggiornamenti dei dispositivi non necessari.

Passaggi dell'implementazione

- Definisci i requisiti di test: stabilisci i requisiti di test ed esegui la pianificazione, ad esempio il tipo di test, i sistemi operativi e il programma di test.
  - Puoi usare Amazon CloudWatch RUM per raccogliere e analizzare i dati lato client e formulare il tuo piano di test.
- Scegli una device farm gestita: seleziona una device farm gestita in grado di supportare i tuoi requisiti di test. Ad esempio, puoi usare AWS Device Farm per testare e comprendere l'impatto delle tue modifiche su un set rappresentativo di hardware.

- Usa l'automazione: utilizza automazione e integrazione continua/implementazione continua (CI/CD) per pianificare ed eseguire i test.
  - Integrazione di Device Farm AWS con la pipeline CI/CD per eseguire i test Selenium sui diversi browser
  - Creazione e test di app iOS e iPadOS con AWS DevOps e servizi mobili
- Rivedi e modifica: revisiona continuamente i risultati dei test e apporta le migliorie necessarie.

Risorse

Documenti correlati:

- Elenco dei dispositivi AWS Device Farm
- Visualizzazione del dashboard CloudWatch RUM

Esempi correlati:

- App di esempio AWS Device Farm per Android
- App di esempio AWS Device Farm per iOS
- Test Appium Web per AWS Device Farm

Video correlati:

- AWS re:Invent 2023 - Improve your mobile and web app quality using AWS Device Farm
- AWS re:Invent 2021 - Optimize applications through end user insights with Amazon CloudWatch RUM

Avvisi

I clienti sono responsabili della propria valutazione autonoma delle informazioni contenute in questo documento. Questo documento: (a) è solo a scopo informativo, (b) mostra le offerte e le pratiche attuali dei prodotti AWS, soggette a modifiche senza preavviso, e (c) non crea alcun impegno o garanzia da parte di AWS e dei suoi affiliati, fornitori o licenziatari. I prodotti o servizi di AWS vengono forniti "così come sono", senza garanzie, rappresentazioni o condizioni di nessun tipo, sia espresse che implicite. Le responsabilità e gli obblighi di AWS verso i propri clienti sono disciplinati dagli accordi AWS e il presente documento non fa parte né modifica alcun accordo tra AWS e i suoi clienti.