AWSWhitepaper

Nozioni di base su più regioni di AWS

Nozioni di base su più regioni di AWS: AWSWhitepaper

Table of Contents

Riassunto e introduzione i
Sintesi 1
Sei tu Well-Architected? 1
Introduzione 1
Progettazione e gestione della resilienza in un'unica regione 3
Nozioni fondamentali su più regioni 1: Comprensione dei requisiti 4
Linee guida chiave 6
Nozioni fondamentali su più regioni 2: Comprendere i dati 7
2a: Comprendere i requisiti di coerenza dei dati 7
2b: Comprendere i modelli di accesso ai dati 8
Guida chiave 10
Nozioni fondamentali su più regioni 3: comprensione delle dipendenze del carico di lavoro 11
3a: servizi AWS 11
3b: dipendenze interne e di terze parti 11
3c: meccanismo di failover 12
3d: dipendenze dalla configurazione 13
Linee guida chiave 13
Principi fondamentali per più regioni 4: prontezza operativa 14
4a: gestione Account AWS 14
4b: Pratiche di implementazione 14
4c: Osservabilità 15
4d: Processi, procedure e test 15
4e: Costo e complessità 16
Guida chiave 17
Conclusioni 18
Collaboratori 19
Approfondimenti 20
Revisioni del documento 21
Note 22
Glossario AWS 23
..... xxiv

Nozioni di base su più regioni di AWS

Data di pubblicazione: 20 dicembre 2022 () Revisioni del documento

Sintesi

Questo paper avanzato di 300 livelli è destinato agli architetti del cloud e ai dirigenti senior AWS che creano carichi di lavoro su cui sono interessati a utilizzare un'architettura multiregionale per migliorare la resilienza dei propri carichi di lavoro. Questo paper presuppone una conoscenza di base dell'AWSinfrastruttura e dei servizi. Descrive i casi d'uso comuni in più regioni, condivide i concetti e le implicazioni fondamentali relativi alla progettazione, allo sviluppo e all'implementazione e fornisce linee guida prescrittive per aiutarti a determinare meglio se un'architettura multiregionale è adatta ai tuoi carichi di lavoro.

Sei tu Well-Architected?

Il AWSWell-Architected Framework ti aiuta a comprendere i pro e i contro delle decisioni che prendi quando crei sistemi nel cloud. I sei pilastri del Framework consentono di apprendere le migliori pratiche architettoniche per progettare e gestire sistemi affidabili, sicuri, efficienti, convenienti e sostenibili. Utilizzando AWS Well-Architected Tool, disponibile gratuitamente in AWS Management Console, puoi esaminare i tuoi carichi di lavoro rispetto a queste best practice rispondendo a una serie di domande per ogni pilastro.

Per ulteriori indicazioni e best practice da parte degli esperti per la tua architettura cloud (implementazioni dell'architettura di riferimento, diagrammi e white paper), consulta l'Architecture Center. AWS

Introduzione

Ciascuna è composta da più zone di disponibilità indipendenti e fisicamente separate all'interno di Regione AWSun'area geografica. Viene mantenuta una rigorosa separazione logica tra i servizi software di ciascuna regione. Questa progettazione mirata garantisce che un guasto dell'infrastruttura o dei servizi in una regione non si traduca in un guasto correlato in un'altra regione.

La maggior parte dei AWS clienti può raggiungere i propri obiettivi di resilienza per un carico di lavoro in una singola regione utilizzando più zone di disponibilità (AZ) o servizi regionali. AWS Tuttavia, un sottoinsieme di clienti utilizza architetture multiregionali per tre motivi.

- Hanno requisiti di elevata disponibilità e continuità delle operazioni per i carichi di lavoro di livello più elevato che ritengono non possano essere soddisfatti in una singola regione.
- Devono soddisfare i requisiti di sovranità dei dati (come il rispetto delle leggi, dei regolamenti e della conformità locali) che richiedono che i carichi di lavoro operino all'interno di una determinata giurisdizione.
- Devono migliorare le prestazioni e l'esperienza del cliente per il carico di lavoro eseguendo i carichi di lavoro nelle sedi più vicine agli utenti finali.

Questo paper si concentra sui requisiti di alta disponibilità e continuità delle operazioni e aiuta a orientarsi tra le considerazioni relative all'adozione di un'architettura multiregionale per un carico di lavoro. Descriviamo i concetti fondamentali che si applicano alla progettazione, allo sviluppo e all'implementazione di un carico di lavoro multiregionale, insieme a un framework prescrittivo per aiutarti a determinare se un'architettura multiregionale è la scelta giusta per un particolare carico di lavoro. Devi assicurarti che un'architettura multiregionale sia la scelta giusta per il tuo carico di lavoro, perché queste architetture sono impegnative ed è possibile che, se non eseguite correttamente, la disponibilità complessiva del carico di lavoro possa diminuire.

Progettazione e gestione della resilienza in un'unica regione

Prima di approfondire i concetti relativi a più regioni, inizia confermando che il carico di lavoro è già il più resiliente possibile in una singola regione. Per raggiungere questo obiettivo, valuta il tuo carico di lavoro rispetto al Reliability Pillar e all'Operational Excellence Pillar del AWS Well-Architected Framework e apporta le modifiche necessarie per adottare le migliori pratiche consigliate. I seguenti concetti sono trattati nel AWS Well-Architected Framework:

- Segmentazione del carico di lavoro basata sui confini del dominio
- Contratti di assistenza ben definiti
- Gestione e accoppiamento delle dipendenze
- Gestione degli errori, dei nuovi tentativi e delle strategie di back-off
- Operazioni idempotenti e transazioni stateful o stateless
- Prontezza operativa e gestione delle modifiche
- Comprendere lo stato del carico di lavoro
- Rispondere agli eventi

Per approfondire ulteriormente la resilienza di una singola regione, rivedi e applica i concetti discussi in Modelli di resilienza Multi-AZ avanzati per la gestione dei guasti grigi. Questo paper fornisce le best practice sull'utilizzo delle repliche in ogni zona di disponibilità per contenere gli errori e approfondisce i concetti Multi-AZ introdotti in Well Architected. AWS Dopo aver applicato appieno i concetti e le best practice consigliati per ottenere la massima resilienza in una singola regione, è possibile valutare un carico di lavoro specifico rispetto ai fondamenti delle architetture multiregionali per determinare se la resilienza del carico di lavoro può essere aumentata utilizzando un approccio multiregionale.

Principi fondamentali per più regioni 1: Comprensione dei requisiti

Come accennato in precedenza, l'elevata disponibilità e la continuità delle operazioni sono ragioni comuni per perseguire architetture multiregionali. Le metriche di disponibilità misurano la percentuale di tempo in cui un carico di lavoro è disponibile per l'uso in un periodo definito, mentre le metriche di continuità delle operazioni misurano il ripristino per eventi su larga scala e in genere di durata maggiore.

La misurazione della disponibilità è un processo quasi continuo. Le misurazioni o le metriche specifiche possono variare, ma in genere si fondono attorno a un obiettivo di disponibilità, spesso definito nove (ad esempio una disponibilità del 99,99%). Con gli obiettivi di disponibilità, un'unica soluzione non va bene per tutti. Gli obiettivi di disponibilità devono essere stabiliti a livello di carico di lavoro anziché applicare un unico obiettivo a tutti i carichi di lavoro, separando i componenti non critici da quelli critici.

Per la continuità delle operazioni, in genere vengono utilizzate le seguenti point-in-time misurazioni:

- Recovery Time Objective (RTO): RTO è il ritardo massimo accettabile tra l'interruzione del servizio e il ripristino del servizio. Questo valore determina una durata accettabile per la quale il servizio è compromesso.
- Recovery Point Objective (RPO): l'RPO è il periodo di tempo massimo accettabile dall'ultimo punto di ripristino dei dati. Ciò determina quella che viene considerata una perdita di dati accettabile tra l'ultimo punto di ripristino e un'interruzione del servizio.

Analogamente alla definizione degli obiettivi di disponibilità, anche RTO e RPO dovrebbero essere definiti a livello di carico di lavoro. Per ottenere una continuità operativa più aggressiva o requisiti di elevata disponibilità, sono necessari maggiori investimenti. Detto questo, non tutte le applicazioni possono richiedere o richiedono lo stesso livello di resilienza. La creazione di un meccanismo di suddivisione in più livelli può aiutare a stabilire il framework per allineare i responsabili aziendali e IT nell'identificazione delle applicazioni più impegnative in base all'impatto aziendale e su livelli adeguati. Nelle tabelle seguenti sono disponibili esempi di suddivisione in più livelli.

Tabella 1 — Esempio di resilienza su più livelli per SLA

Contratto sul livello di servizio (SLA) di disponibilità	Livello di resilienza	Tempo di inattività accettabile/anno
99,99%	Platino	52,60 minuti
99,90%	Oro	8,77 ore
99,5%	Argento	1,83 giorni

Tabella 2 — Esempio di resilienza su più livelli per RTO e RPO

Livello	RTO massimo	RPO massimo	Criteri	Costo
Platino	15 minuti	cinque minuti	Carichi di lavoro mission-critical	$$$
Oro	15 minuti — sei ore	due ore	Carichi di lavoro importanti, ma non cruciali	$$
Argento	sei ore, pochi giorni	24 ore	Carichi di lavoro non critici	$

Quando si progettano carichi di lavoro per la resilienza, è necessaria una comprensione della relazione tra alta disponibilità e continuità delle operazioni. Ad esempio, se un carico di lavoro richiede una disponibilità del 99,99%, non sono tollerabili più di 53 minuti di inattività all'anno. Possono essere necessari almeno cinque minuti per rilevare un guasto e altri dieci minuti prima che un operatore interagisca, prenda decisioni sulle fasi di ripristino ed esegua queste operazioni. Non è insolito che il ripristino di un singolo problema richieda dai 30 ai 45 minuti. In questo caso, una strategia multiregionale che preveda un'istanza isolata in grado di rimuovere gli impatti correlati può consentire la continuità delle operazioni grazie al failover entro un periodo di tempo limitato e alla valutazione indipendente del danno iniziale. È qui che è necessario definire l'RTO e l'RPO appropriati.

Per i carichi di lavoro mission-critical che hanno esigenze di disponibilità estreme (ad esempio, disponibilità del 99,99% o superiore) o requisiti di continuità operativa rigorosi che possono essere soddisfatti solo eseguendo il failover in un'altra regione, potrebbe essere appropriato un approccio

multiregionale. Tuttavia, questi requisiti sono in genere applicabili solo a un piccolo sottoinsieme del portafoglio di carichi di lavoro di un'azienda con un tempo di ripristino limitato, misurato in minuti o ore. A meno che un'applicazione non richieda un tempo di ripristino di pochi minuti o poche ore, attendere che un'interruzione regionale dell'applicazione venga risolta all'interno della regione interessata potrebbe essere un approccio migliore, in genere in linea con carichi di lavoro di livello inferiore.

Prima di implementare un'architettura multiregionale, i responsabili delle decisioni aziendali e i team tecnici devono essere allineati sulle implicazioni in termini di costi, compresi i fattori di costo operativi e infrastrutturali. Una tipica architettura multiregionale può comportare un aumento dei costi due volte maggiore rispetto a un approccio a regione singola. Sebbene esistano diversi modelli multiregionali per la continuità aziendale, ad esempio l'utilizzo di hot standby, warm standby e luce pilota, il modello con il rischio più basso di raggiungere gli obiettivi di ripristino comporterà l'utilizzo di hot standby e raddoppierà il costo del carico di lavoro.

Linee guida chiave

- Gli obiettivi di disponibilità e continuità delle operazioni, come RTO e RPO, devono essere stabiliti per carico di lavoro e allineati agli stakeholder aziendali e IT.
- La maggior parte degli obiettivi di disponibilità e continuità delle operazioni può essere raggiunta all'interno di una singola regione. Per quanto riguarda gli obiettivi che non possono essere raggiunti con una singola regione, è opportuno prendere in considerazione più regioni, con una visione chiara dei compromessi tra costi, complessità e benefici.

Principi fondamentali per più regioni 2: comprensione dei dati

La gestione dei dati è un problema non banale con le architetture multiregionali. La distanza geografica tra le regioni impone una latenza inevitabile, che si manifesta con il tempo necessario per replicare i dati tra le regioni. Saranno necessari compromessi tra disponibilità, coerenza dei dati e introduzione di ordini di grandezza di latenza più elevati in un carico di lavoro che utilizza un'architettura multiregionale. Che si utilizzi la replica asincrona o sincrona, sarà necessario modificare l'applicazione per gestire i cambiamenti comportamentali imposti dalla tecnologia di replica. È molto difficile prendere un'applicazione esistente progettata per una singola regione e renderla multiregionale a causa delle sfide legate alla coerenza e alla latenza dei dati. Comprendere i requisiti di coerenza dei dati e i modelli di accesso ai dati per carichi di lavoro particolari è fondamentale per valutare i compromessi.

2a: Comprendere i requisiti di coerenza dei dati

Il teorema CAP fornisce un riferimento per ragionare sui compromessi tra coerenza dei dati, disponibilità e partizioni di rete, di cui solo due possono essere soddisfatte contemporaneamente per un carico di lavoro. La multiregione per definizione include le partizioni di rete tra regioni, quindi è necessario scegliere tra disponibilità e coerenza.

Se si seleziona la disponibilità dei dati tra le regioni, non si verificherà una latenza significativa durante le scritture transazionali, poiché si fa affidamento sulla replica asincrona dei dati impegnati tra le regioni, con conseguente riduzione della coerenza tra le regioni fino al completamento della replica. Con la replica asincrona, in caso di errore nella regione principale, c'è un'alta probabilità di scritture in attesa della replica dalla regione principale. Ciò porta a uno scenario in cui i dati più recenti non sono disponibili fino alla ripresa della replica ed è necessario un processo di riconciliazione per gestire le transazioni in corso che non sono state replicate dalla regione in cui si è verificata l'interruzione.

Per i carichi di lavoro in cui è preferita la replica asincrona, puoi utilizzare servizi come Amazon Aurora e Amazon DynamoDB, che forniscono la replica asincrona tra regioni. Sia le tabelle globali Amazon Aurora Global Database che Amazon DynamoDB dispongono di parametri CloudWatchAmazon predefiniti per facilitare il monitoraggio del ritardo di replica.

Progettare il carico di lavoro per sfruttare le architetture basate sugli eventi è un vantaggio per una strategia multiregionale, perché significa che il carico di lavoro può includere la replica asincrona

dei dati e consente la ricostruzione dello stato mediante la riproduzione degli eventi. Poiché i servizi di streaming e messaggistica memorizzano nel buffer i dati del payload dei messaggi in un'unica regione, un processo di failover/failback regionale deve includere un meccanismo per reindirizzare i flussi di dati di input dei client, nonché per riconciliare i payload in volo e/o non consegnati archiviati nella regione in cui si è verificata l'interruzione.

Se si seleziona la coerenza, si verificherà una latenza significativa poiché i dati vengono replicati in modo sincrono durante le scritture transazionali. Quando si scrive in più regioni in modo sincrono, se la scrittura non riesce in tutte le regioni, la disponibilità è potenzialmente ridotta perché la transazione non verrà confermata e dovrà essere ritentata. I nuovi tentativi di scrivere i dati in tutte le regioni in modo sincrono vengono eseguiti a scapito della latenza ad ogni tentativo. A un certo punto, quando i nuovi tentativi saranno esauriti, sarà necessario decidere se annullare completamente la transazione, riducendo così la disponibilità, oppure affidarla solo alle regioni disponibili, con conseguenti incongruenze. Esistono tecnologie di formazione del quorum come Paxos, che possono aiutare a replicare e inviare dati in modo sincrono, ma che richiedono investimenti significativi da parte degli sviluppatori.

Quando le scritture prevedono la replica sincrona su più regioni per soddisfare elevati requisiti di coerenza, la latenza di scrittura aumenta di un ordine di grandezza. Una latenza di scrittura più elevata non è qualcosa che in genere può essere adattata a posteriori in un'applicazione senza modifiche significative. Idealmente, deve essere presa in considerazione quando l'applicazione viene progettata per la prima volta. Per i carichi di lavoro multiregionali in cui la replica sincrona è una priorità, le soluzioni dei AWSpartner possono essere d'aiuto.

2b: Comprensione dei modelli di accesso ai dati

I modelli di accesso ai dati dei carichi di lavoro rientrano in uno dei seguenti tipi: intensivo in lettura o in scrittura. La comprensione di questa caratteristica per un particolare carico di lavoro guiderà la selezione di un'architettura multiregionale appropriata.

Per carichi di lavoro ad alta intensità di lettura, come i contenuti statici completamente di sola lettura, è possibile realizzare un'architettura multiregionale attiva/attiva senza complessità significative. La distribuzione di contenuti statici all'edge tramite una rete di distribuzione dei contenuti (CDN) garantisce la disponibilità memorizzando nella cache i contenuti più vicini all'utente finale; l'utilizzo di set di funzionalità come il failover Origin all'interno di Amazon CloudFront può contribuire a raggiungere questo obiettivo. Un'altra opzione consiste nell'implementare l'elaborazione stateless in più regioni e utilizzare il DNS per indirizzare gli utenti alla regione più vicina per leggere il contenuto. A tal fine è possibile utilizzare Route 53 con politica di routing di geolocalizzazione.

Per carichi di lavoro ad alta intensità di lettura che hanno una percentuale maggiore di letture rispetto alle scritture, è possibile utilizzare una strategia globale di lettura locale e scrittura. Ciò implica che tutte le scritture vadano a un database in una regione specifica con replica asincrona dei dati in tutte le altre regioni e a tal fine le letture possono essere eseguite in qualsiasi regione. Questo approccio richiede un carico di lavoro tale da garantire la coerenza finale, in quanto le letture locali potrebbero risultare obsolete a causa della maggiore latenza per la replica delle scritture tra regioni diverse.

Aurora Global Database può aiutare a fornire repliche di lettura in una regione di standby in grado di gestire esclusivamente tutto il traffico di lettura a livello locale e un singolo datastore primario in una regione specifica per gestire le scritture. I dati vengono replicati in modo asincrono dai database primari a quelli di standby (Read Repliche) e i database di standby possono essere promossi a primari se è necessario eseguire il failover delle operazioni nella regione di standby. Se un carico di lavoro è più adatto per modelli di dati non relazionali, DynamoDB può essere utilizzato anche in questo approccio. Anche in questo caso, il carico di lavoro deve garantire la coerenza finale, il che potrebbe richiedere una riscrittura se non è stato progettato per questo scopo fin dall'inizio.

Per i carichi di lavoro ad alta intensità di scrittura, è necessario selezionare una regione principale e incorporare nel carico di lavoro la capacità di failover su una regione di standby. Rispetto a un approccio attivo/attivo, un approccio primario/standby è meno complicato. Questo perché per un'architettura attiva/attiva, il carico di lavoro dovrà essere riscritto per gestire il routing intelligente verso le regioni, stabilire l'affinità delle sessioni, garantire transazioni idempotenti e gestire potenziali conflitti.

La maggior parte dei carichi di lavoro che utilizzano la resilienza multiregionale non richiederà un approccio attivo/attivo. È possibile utilizzare una strategia di sharding per fornire una maggiore resilienza limitando il raggio d'azione di un danno all'interno della base clienti. Se è possibile suddividere in modo efficace una base di clienti, è possibile selezionare diverse regioni primarie per ogni shard. Ad esempio, se è possibile suddividere i client in modo che metà dei client siano allineati alla Regione Uno e l'altra metà alla Regione Due, trattando le regioni come celle, è possibile creare un approccio cellulare multiregionale, che si traduce in una riduzione del raggio di impatto del carico di lavoro.

L'approccio di sharding può essere combinato con un approccio primario/standby per fornire funzionalità di failover per gli shard. Sarà necessario integrare nel carico di lavoro un processo di failover testato e un processo di riconciliazione dei dati per garantire la coerenza transazionale degli archivi di dati dopo il failover. Questi sono trattati più dettagliatamente più avanti in questo paper.

Linee guida chiave

- È molto probabile che le scritture in sospeso per la replica non vengano salvate nella regione di standby in caso di errore. I dati non saranno disponibili fino alla ripresa della replica (presupponendo la replica asincrona).
- Come parte del failover, sarà necessario un processo di riconciliazione dei dati per garantire il mantenimento di uno stato transazionale coerente per i datastore che utilizzano la replica asincrona.
- Quando è richiesta una forte coerenza, sarà necessario modificare i carichi di lavoro per tollerare la latenza richiesta del datastore che si replica in modo sincrono.

Nozioni fondamentali su più regioni 3: Comprendere le dipendenze del carico di lavoro

Un carico di lavoro specifico può avere diverse dipendenze in una regione, ad esempio AWS servizi utilizzati, dipendenze interne, dipendenze di terze parti, dipendenze di rete, certificati, chiavi, segreti e parametri. Per garantire il funzionamento del carico di lavoro durante uno scenario di errore, non dovrebbero esserci dipendenze tra la regione principale e la regione di standby; ciascuna dovrebbe essere in grado di funzionare indipendentemente l'una dall'altra. A tal fine, è necessario esaminare attentamente tutte le dipendenze del carico di lavoro per garantire che siano disponibili all'interno di ciascuna regione. Ciò è necessario perché un errore nella regione principale non dovrebbe avere un impatto nella regione di standby. Inoltre, è fondamentale conoscere il funzionamento del carico di lavoro quando una dipendenza si trova in uno stato degradato o completamente non disponibile, in modo da poter progettare soluzioni per gestirla in modo appropriato.

3a: servizi AWS

Quando si progetta un'architettura multiregionale, è necessaria una comprensione dei AWS servizi specifici che verranno utilizzati. Il primo aspetto è capire quali funzionalità dispone il servizio per abilitare più regioni e se è necessario progettare una soluzione per raggiungere gli obiettivi multiregionali. Ad esempio, con Amazon Aurora e Amazon DynamoDB, è disponibile una funzionalità per replicare in modo asincrono i dati in una regione di standby. Qualsiasi dipendenza dal AWS servizio dovrà essere disponibile in tutte le regioni da cui verrà eseguito un carico di lavoro. Per garantire che i servizi che verranno utilizzati siano disponibili nelle regioni desiderate, consulta l'Regione AWSelenco dei servizi.

3b: Dipendenze interne e di terze parti

Per tutte le dipendenze interne di un carico di lavoro, assicurati che sia disponibile nelle regioni da cui verrà utilizzato il carico di lavoro. Ad esempio, se il carico di lavoro è composto da molti microservizi, informati su tutti i microservizi che costituiscono una funzionalità aziendale. Da lì, assicurati che tutti questi microservizi siano distribuiti in ogni regione in cui il carico di lavoro funzionerà.

Le chiamate interregionali tra microservizi all'interno di un carico di lavoro non sono consigliate e l'isolamento regionale dovrebbe essere mantenuto. Questo perché la creazione di dipendenze tra regioni aumenta il rischio di errori correlati, il che annulla i vantaggi che si stanno cercando di ottenere con implementazioni regionali isolate del carico di lavoro. Anche le dipendenze locali

potrebbero far parte del carico di lavoro, quindi è fondamentale comprendere come le caratteristiche di queste integrazioni potrebbero cambiare se la regione principale dovesse cambiare. Ad esempio, se la regione di standby si trova più lontana dall'ambiente locale, l'aumento della latenza avrà un impatto negativo.

Comprendere le soluzioni Software as a Service (SaaS), i kit di sviluppo software (SDK) e altre dipendenze da prodotti di terze parti e la possibilità di utilizzare scenari in cui tali dipendenze sono degradate o non disponibili forniranno maggiori informazioni su come la catena di sistemi opera e si comporta in diverse modalità di errore. Queste dipendenze possono rientrare in un codice applicativo, dal modo in cui i segreti vengono gestiti esternamente utilizzando AWS Secrets Manager o una soluzione di vault di terze parti (come Hashicorp), ai sistemi di autenticazione che dipendono da IAM Identity Center per gli accessi federati.

La ridondanza quando si tratta di dipendenze può contribuire ad aumentare la resilienza. Esiste anche la possibilità che una soluzione SaaS o una dipendenza di terze parti utilizzi lo stesso carico di lavoro primarioRegione AWS. In tal caso, è necessario collaborare con il fornitore per determinare se il suo livello di resilienza corrisponde ai requisiti per il carico di lavoro.

Inoltre, tieni presente il destino condiviso tra il carico di lavoro e le sue dipendenze, ad esempio le applicazioni di terze parti. Se le dipendenze non sono disponibili in (o da) una regione secondaria dopo un failover, il carico di lavoro potrebbe non essere ripristinato completamente.

3c: meccanismo di failover

Il Domain Name System (DNS) viene comunemente utilizzato come meccanismo di failover per spostare il traffico dalla regione principale a una regione di standby. Esamina e analizza in modo critico tutte le dipendenze assunte dal meccanismo di failover. Ad esempio, se il tuo carico di lavoro utilizza Amazon Route 53, sapere che il piano di controllo è ospitato negli Stati Uniti orientali 1 significa che stai assumendo una dipendenza dal piano di controllo in quella regione specifica. Questa operazione non è consigliata come parte di un meccanismo di failover se anche la regione principale è US-East-1. Se si utilizza un altro meccanismo di failover, è necessaria una conoscenza approfondita di qualsiasi scenario in cui non funzionerebbe come previsto. Una volta stabilita questa comprensione, pianificate gli imprevisti o sviluppate un nuovo meccanismo, se necessario. Consulta la sezione Creazione di meccanismi di disaster recovery con Amazon Route 53 per scoprire gli approcci che puoi utilizzare per eseguire correttamente il failover.

Come discusso nella sezione sulle dipendenze interne, tutti i microservizi che fanno parte di una funzionalità aziendale devono essere disponibili in ogni regione in cui viene distribuito il carico di

lavoro. Nell'ambito della strategia di failover, le funzionalità aziendali devono essere integrate per eliminare la possibilità di chiamate tra regioni diverse. In alternativa, se i microservizi effettuano il failover in modo indipendente, ciò comporta il rischio di un comportamento indesiderato, in cui i microservizi possono effettuare chiamate tra regioni diverse, con conseguente latenza e l'indisponibilità del carico di lavoro in caso di timeout del client.

3d: dipendenze di configurazione

I certificati, le chiavi, i segreti e i parametri fanno parte dell'analisi delle dipendenze necessaria durante la progettazione per più regioni. Quando possibile, è meglio localizzare questi componenti all'interno di ciascuna regione in modo che non abbiano un destino condiviso tra le regioni per queste dipendenze. Per quanto riguarda i certificati, la scadenza dovrebbe variare da un paese all'altro e, se possibile, da una regione all'altra, per evitare che un certificato in scadenza (con allarmi impostati per notificare in anticipo) influisca su più regioni.

Anche le chiavi e i segreti di crittografia devono essere specifici della regione. In questo modo, se si verifica un errore nella rotazione di una chiave o di un segreto, l'impatto è limitato a una regione specifica.

Infine, tutti i parametri del carico di lavoro devono essere archiviati localmente affinché il carico di lavoro possa essere recuperato nella regione specifica.

Linee guida chiave

- Un'architettura multiregionale trae vantaggio dalla separazione fisica e logica tra le regioni. L'introduzione di dipendenze interregionali a livello di applicazione annulla questo vantaggio. Evita tali dipendenze.
- I controlli di failover dovrebbero funzionare senza dipendenze dalla regione principale.
- È necessario coordinare il failover a livello di capacità aziendale per eliminare la possibilità di un aumento della latenza e della dipendenza delle chiamate interregionali.

Principi fondamentali per più regioni 4: prontezza operativa

La gestione di un carico di lavoro multiregionale è un'attività complessa che comporta sfide operative specifiche per più regioni. Queste includono la Account AWS gestione, la riorganizzazione dei processi di implementazione, la creazione di una strategia di osservabilità multiregionale, la creazione e il test dei runbook di failover e failback e quindi la gestione dei costi. L'Operational Readiness Review (ORR) può aiutare i team a preparare un carico di lavoro per la produzione, indipendentemente dal fatto che venga eseguito in una singola regione o in più regioni.

4a: gestione Account AWS

Per distribuire un carico di lavoro in tutte le regioniRegioni AWS, assicurati che tutte le quote di AWS servizio all'interno di un account siano uguali tra le regioni. Innanzitutto, scopri tutti i AWS servizi che fanno parte dell'architettura, esamina l'utilizzo pianificato nelle regioni di standby, quindi confrontali con l'utilizzo attuale. In alcuni casi, se la regione di standby non è mai stata utilizzata in precedenza, puoi fare riferimento alle quote di servizio predefinite per comprendere il punto di partenza. Quindi, per tutti i servizi che verranno utilizzati, richiedi un aumento della quota utilizzando la console Service Quotas (accesso richiesto) o le API.

AWSI ruoli di Identity and Access Management (IAM) devono essere configurati in ogni regione per garantire che gli operatori, gli strumenti di automazione e AWS i servizi dispongano delle autorizzazioni appropriate per le risorse all'interno della regione di standby. L'isolamento regionale dei ruoli consente di ottenere l'isolamento regionale che perseguiamo per le architetture multiregionali. Assicurati che queste autorizzazioni siano disponibili prima di passare alla modalità attiva con una regione in standby.

4b: Pratiche di implementazione

Con funzionalità multiregionali, l'implementazione del carico di lavoro in più regioni può essere complessa. AWS CloudFormationaiuta a implementare l'infrastruttura in una o più regioni e può essere personalizzato in base alle esigenze. AWS CodePipelineaiuta a fornire una pipeline di integrazione/distribuzione continua (CI/CD) quasi continua, che prevede azioni interregionali che consentono l'implementazione in regioni diverse dalla regione in cui si trova la pipeline. Questo, combinato con solide strategie di implementazione come blue/green, consente un'implementazione con tempi di inattività minimi o pari a zero.

Tuttavia, l'implementazione delle funzionalità stateful può essere più complessa quando lo stato dell'applicazione o dei dati non è esternalizzato in un archivio persistente. In queste situazioni, personalizza attentamente il processo di implementazione in base alle tue esigenze. Progetta la pipeline e il processo di distribuzione in modo da distribuirla in una regione alla volta, anziché in più regioni contemporaneamente. Ciò riduce la possibilità di guasti correlati tra le regioni. Per conoscere le tecniche utilizzate da Amazon per automatizzare le distribuzioni di software, leggi l'articolo di Builder Library Automating safe hands-off deployments.

4c: Osservabilità

Quando progetti per più regioni, considera come verrà monitorata la salute di tutti i componenti di ciascuna regione per ottenere una visione olistica della salute regionale. Ciò potrebbe includere il monitoraggio delle metriche per il ritardo di replica, che non viene preso in considerazione per il carico di lavoro di una singola regione.

Quando crei un'architettura multiregionale, prendi in considerazione l'osservazione delle prestazioni del carico di lavoro anche nelle regioni di standby. Ciò include il controllo dello stato di salute e l'esecuzione di canarini (test sintetici) dalla regione di attesa, in modo da avere una visione esterna dello stato di salute del primario. Inoltre, puoi utilizzare Amazon CloudWatch Internet Monitor per comprendere lo stato della rete esterna e le prestazioni dei tuoi carichi di lavoro dal punto di vista dell'utente finale. Analogamente, la regione principale dovrebbe avere la stessa osservabilità per monitorare la regione di standby. Questi canarini dovrebbero monitorare le metriche relative all'esperienza dei clienti per verificare lo stato generale del carico di lavoro. Ciò è necessario perché se si verificasse un problema nella regione primaria, l'osservabilità nella regione primaria potrebbe essere compromessa e influirebbe sulla capacità di valutare lo stato del carico di lavoro.

In tal caso, osservare al di fuori di quella regione può fornire informazioni. Queste metriche devono essere inserite nelle dashboard disponibili in ogni regione e gli allarmi devono essere creati in ciascuna regione. Poiché Amazon CloudWatch è un servizio regionale, la disponibilità di tali servizi in entrambe le regioni è un requisito. Questi dati di monitoraggio verranno utilizzati per effettuare la chiamata al failover da una regione principale a una regione di standby.

4d: Processi, procedure e test

Il momento migliore per rispondere alla domanda «Quando devo effettuare il failover?» è molto prima che sia necessario. I piani di continuità aziendale comprensivi di personale, processi e tecnologia devono essere tutti definiti con largo anticipo rispetto al problema e testati regolarmente. Decidi

un quadro decisionale di recupero. Se esiste un processo di ripristino ben collaudato e i tempi necessari per il ripristino sono ben compresi, è possibile scegliere il momento in cui avviare il processo di ripristino che soddisfi l'obiettivo RTO attraverso un failover. Questo momento potrebbe avvenire immediatamente dopo l'identificazione di un problema con l'applicazione nella regione principale, oppure potrebbe avvenire ulteriormente in un evento in cui le opzioni di ripristino all'interno dell'applicazione nella regione sono state esaurite e dovrebbe ora iniziare un failover per soddisfare l'RTO.

Sebbene l'azione di failover stessa debba essere automatizzata al 100%, la decisione di attivarla dovrebbe essere presa da un essere umano (in genere un numero limitato di individui predeterminati all'interno dell'organizzazione). Inoltre, i criteri per decidere in merito a un failover devono essere chiaramente definiti e compresi a livello globale con l'organizzazione. Questi processi possono essere definiti e completati utilizzando i runbook di AWS System Manager, che consentono end-to-end l'automazione completa e garantiscono la coerenza dell'esecuzione del processo durante i test e il failover.

Questi runbook devono essere disponibili nella regione principale e nella regione di standby per avviare i processi di failover o failback. Una volta implementata questa automazione, è necessario definire e seguire una cadenza di test regolare. Ciò garantisce che, quando si verifica un evento reale, la risposta avvenga secondo un processo ben definito e pratico in cui l'organizzazione ha fiducia. È inoltre importante tenere a mente le tolleranze stabilite per i processi di riconciliazione dei dati. Conferma che i requisiti RPO/RTO stabiliti siano soddisfatti con il processo proposto.

4e: Costo e complessità

Le implicazioni in termini di costi di un'architettura multiregionale sono determinate da un maggiore utilizzo dell'infrastruttura, dal sovraccarico operativo e dal tempo impiegato per le risorse. Come accennato in precedenza, il costo dell'infrastruttura in una regione di standby è simile al costo dell'infrastruttura in una regione primaria durante il pre-provisioning, il che significa che il costo è due volte superiore. Fornisci capacità in modo che sia sufficiente per le operazioni quotidiane, riservando comunque una capacità di buffer sufficiente per tollerare i picchi di domanda e configura gli stessi limiti in ogni regione.

Inoltre, potrebbero essere necessarie modifiche a livello di applicazione per funzionare correttamente in un'architettura multiregionale se si adotta un'architettura active-active, che può richiedere molto tempo e risorse per la progettazione e il funzionamento. Come minimo, le organizzazioni dovrebbero dedicare del tempo alla comprensione delle dipendenze tecniche e commerciali in ciascuna regione e alla progettazione di processi di failover e failback.

I team dovrebbero inoltre sottoporsi ai normali esercizi di failover e failback per sentirsi a proprio agio con i runbook che verranno utilizzati durante un evento. Sebbene siano incredibilmente importanti e cruciali per ottenere i risultati attesi da un investimento in più regioni, questi esercizi rappresentano un costo-opportunità e sottraggono tempo e risorse ad altre attività.

Linee guida chiave

- AWSle quote di servizio devono essere riviste e mantenute paritarie in tutte le regioni in cui si svolgerà il carico di lavoro.
- Il processo di implementazione dovrebbe riguardare una regione alla volta, anziché più regioni contemporaneamente.
- È necessario monitorare parametri aggiuntivi, come il ritardo di replica, e sono specifici per gli scenari multiregionali.
- Estendi il monitoraggio del carico di lavoro oltre la regione principale. Le metriche relative all'esperienza del cliente devono essere monitorate per regione e misurate al di fuori di ciascuna regione in cui è in esecuzione un carico di lavoro.
- Il failover e il failback devono essere testati regolarmente. Garantite l'implementazione di un unico runbook per i processi di failover e failback da utilizzare sia durante i test che durante un evento dal vivo. I runbook per i test e gli eventi live non possono essere diversi.

Conclusioni

Questo white paper illustra i casi d'uso più comuni per più regioni, i fondamenti su come implementare un'architettura multiregionale e le implicazioni di questo approccio. Questi principi fondamentali possono essere applicati a qualsiasi carico di lavoro e utilizzati come framework per aiutare a decidere se un'architettura multiregionale sia o meno l'approccio giusto per una particolare azienda.

Collaboratori

Hanno collaborato alla stesura del presente documento:

Collaboratore tecnico:

- John Formento, Jr., responsabile delle soluzioni, team multiregionale AWS

Collaboratore editoriale:

- Lisi Lewis, Senior Manager, Marketing dei prodotti

Approfondimenti

Per ulteriori informazioni, fare riferimento a:

- Modelli di resilienza Multi-AZ avanzati (white paper) AWS
- Pilastro dell'affidabilità - AWS Well-Architected Framework
- Disponibilità e oltre: comprensione e miglioramento della resilienza dei sistemi distribuiti su (white paper) AWS AWS
- AWSLimiti di isolamento dei guasti (white paper) AWS

Revisioni del documento

Per ricevere notifiche sugli aggiornamenti di questo white paper, iscriviti al feed RSS.

Modifica	Descrizione	Data
Documento pubblicato	Prima pubblicazione.	20 dicembre 2022

Note

I clienti sono responsabili della propria valutazione indipendente delle informazioni contenute in questo documento. Questo documento: (a) è solo a scopo informativo, (b) rappresenta le attuali offerte e pratiche dei prodotti AWS, che sono soggette a modifiche senza preavviso, e (c) non costituisce alcun impegno o garanzia da parte di AWS e dei suoi affiliati, fornitori o licenziatari. I prodotti o i servizi AWS sono forniti così come sono, senza garanzie, dichiarazioni o condizioni di alcun tipo, sia esplicite che implicite. Le responsabilità di AWS nei confronti dei propri clienti sono definite dai contratti AWS e il presente documento non costituisce parte né modifica qualsivoglia contratto tra AWS e i suoi clienti.

Glossario AWS

Per la terminologia AWS più recente, consultare il glossario AWS nella documentazione di riferimento per Glossario AWS.

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.