Whitepaper AWS

Hosting di applicazioni Web in AWS Cloud

Hosting di applicazioni Web in AWS Cloud: Whitepaper AWS

Table of Contents

Riassunto 1
Riassunto 1
Panoramica dell'hosting Web tradizionale 2
Hosting di applicazioni Web nel cloud con AWS 4
Utilità di AWS per risolvere i problemi comuni di hosting di applicazioni Web 4
Alternativa conveniente ai parchi istanze sovradimensionati necessari per gestire i picchi 4
Soluzione scalabile per gestire i picchi di traffico imprevisti 5
Soluzione on demand per ambienti di test, carico, beta e riproduzione 5
Architettura di AWS Cloud per l'hosting Web 5
Componenti chiave di un'architettura di hosting Web AWS 7
Gestione delle reti 7
Distribuzione di contenuti 8
Gestione dei nomi DNS pubblici 8
Sicurezza dell'host 9
Bilanciamento del carico tra cluster 9
Ricerca di altri host e servizi 9
Memorizzazione nella cache all'interno dell'applicazione Web 10
Configurazione, backup e failover del database 10
Archiviazione e backup di dati e risorse 13
Scalabilità automatica del parco istanze 14
Caratteristiche di sicurezza aggiuntive 14
Failover con AWS 15
Considerazioni chiave per l'uso di AWS per l'hosting Web 17
Nessuna appliance di rete fisica 17
Firewall ovunque 17
Considerazioni sulla disponibilità di più data center 17
Host effimeri e dinamici 18
Considerazioni su container ed elaborazione serverless 18
Considerazioni sulla distribuzione automatica 18
Conclusione e collaboratori 20
Conclusione 20
Collaboratori 20
Approfondimenti 21
Revisioni del documento 22

Avvisi 24

Hosting di applicazioni Web in AWS Cloud

Data di pubblicazione: 20 agosto 2021 (Revisioni del documento)

Riassunto

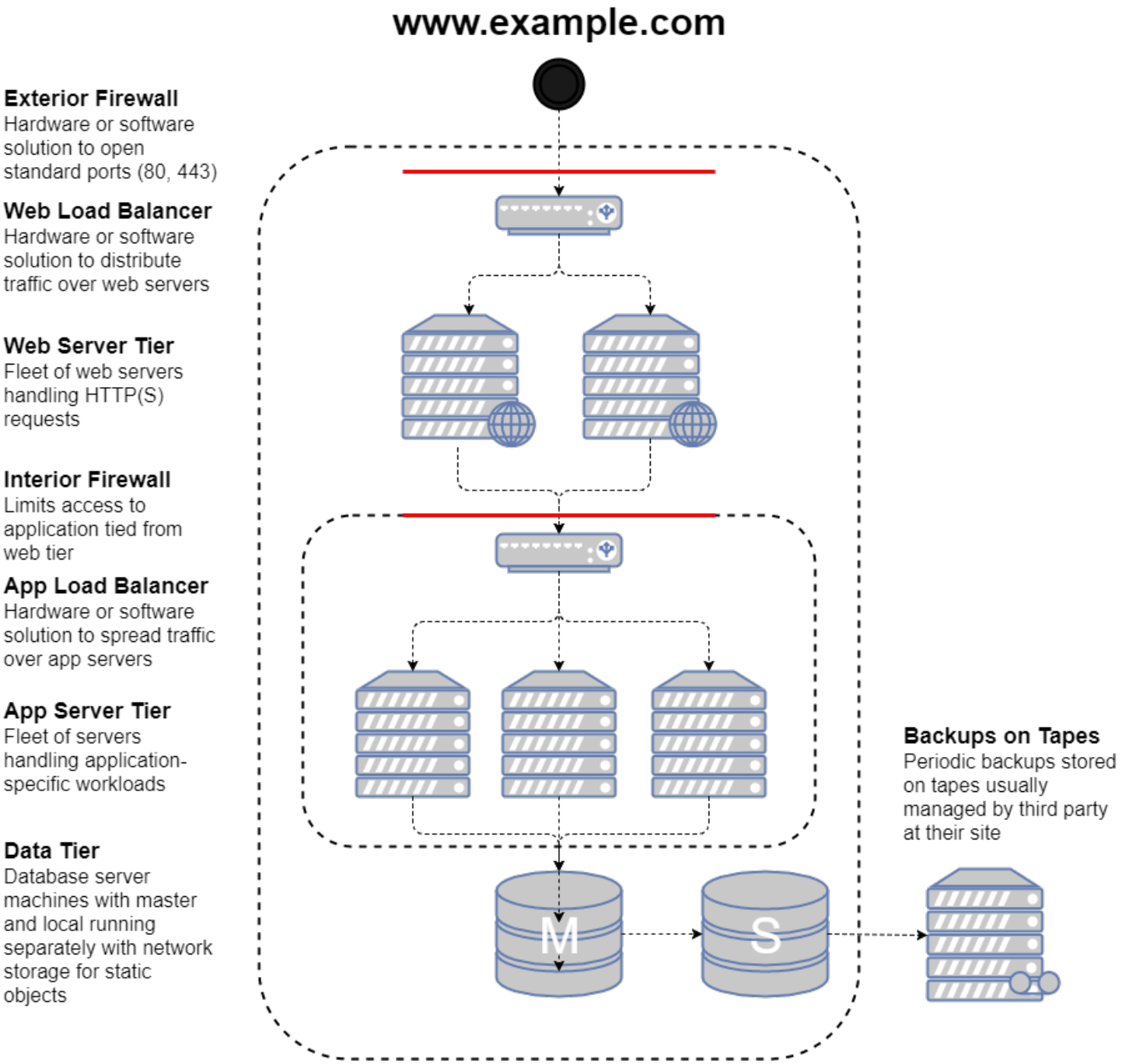
Le architetture Web On-Premise tradizionali richiedono soluzioni complesse e previsioni accurate della capacità riservata per garantire l'affidabilità. I periodi di traffico di picco intensi e le oscillazioni incontrollate nei modelli di traffico hanno come conseguenza bassi tassi di utilizzo di hardware costoso. Ciò comporta costi operativi elevati per mantenere l'hardware inattivo e un uso non efficiente del capitale per l'hardware sottoutilizzato.

Amazon Web Services (AWS) fornisce un'infrastruttura affidabile, scalabile, sicura e a prestazioni elevate per le applicazioni Web più complesse. I costi IT di questa infrastruttura dipendono dai modelli di traffico dei clienti in tempo quasi reale.

Questo whitepaper è destinato ai responsabili IT e ai progettisti di sistemi che desiderano capire come eseguire architetture Web tradizionali nel cloud per ottenere elasticità, scalabilità e affidabilità.

Panoramica dell'hosting Web tradizionale

L'hosting Web scalabile è un aspetto notoriamente problematico. La figura seguente mostra un'architettura di hosting Web tradizionale che implementa un modello di applicazione Web a tre livelli comune. In questo modello l'architettura è suddivisa nei livelli di presentazione, applicazione e persistenza. La scalabilità viene fornita aggiungendo host a questi livelli. L'architettura offre anche caratteristiche integrate di prestazioni, failover e disponibilità. L'architettura di hosting Web tradizionale può essere facilmente trasferita in AWS Cloud con poche modifiche.



Architettura di hosting Web tradizionale

Le sezioni seguenti illustrano perché e come è consigliabile ed è possibile implementare un'architettura di questo tipo in AWS Cloud.

Hosting di applicazioni Web nel cloud con AWS

La prima domanda da porsi riguarda il valore che è possibile ottenere dallo spostamento di una soluzione classica di hosting di applicazioni Web in AWS Cloud. Dopo aver deciso che il cloud è la scelta giusta, è necessaria un'architettura adeguata. Questa sezione aiuta a valutare una soluzione AWS Cloud. La distribuzione dell'applicazione Web nel cloud viene messa a confronto con una distribuzione On-Premise, viene presenta un'architettura AWS Cloud per l'hosting dell'applicazione e vengono illustrati i componenti chiave della soluzione di architettura di AWS Cloud.

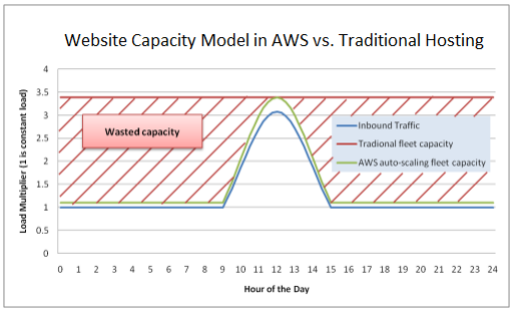
Utilità di AWS per risolvere i problemi comuni di hosting di applicazioni Web

Se sei responsabile dell'esecuzione di un'applicazione Web, potresti trovarti ad affrontare una serie di problemi di infrastruttura e architettura per i quali AWS può fornire soluzioni semplici e convenienti. Di seguito sono illustrati alcuni vantaggi dell'uso di AWS rispetto a un modello di hosting tradizionale.

Alternativa conveniente ai parchi istanze sovradimensionati necessari per gestire i picchi

Nel modello di hosting tradizionale, è necessario effettuare il provisioning di server per gestire la capacità di picco. I cicli inutilizzati sono sprecati al di fuori dei periodi di picco. Le applicazioni Web ospitate da AWS possono usufruire del provisioning on demand di server aggiuntivi, in modo da poter adattare continuamente la capacità e i costi ai modelli di traffico effettivi.

Il grafico seguente mostra ad esempio un'applicazione Web con un picco di utilizzo dalle 9:00 alle 15:00 e un utilizzo inferiore per il resto della giornata. Un approccio con scalabilità automatica in base alle tendenze effettive del traffico, che prevede il provisioning delle risorse solo quando necessario, comporterebbe un minor spreco di capacità e una riduzione dei costi superiore al 50%.



Esempio di spreco di capacità in un modello di hosting classico

Soluzione scalabile per gestire i picchi di traffico imprevisti

Una conseguenza più grave del provisioning lento associato a un modello di hosting tradizionale è l'incapacità di rispondere tempestivamente a picchi di traffico imprevisti. Capita spesso che le applicazioni Web smettano di essere disponibili a causa di un picco inaspettato di traffico dopo che il sito è stato menzionato nei media più popolari. In AWS Cloud la stessa capacità on demand che permette il dimensionamento delle applicazioni Web in base ai picchi di traffico permette anche di gestire un carico imprevisto. È possibile avviare nuovi host, che sono disponibili in pochi minuti e possono essere disconnessi altrettanto rapidamente quando il traffico torna alla normalità.

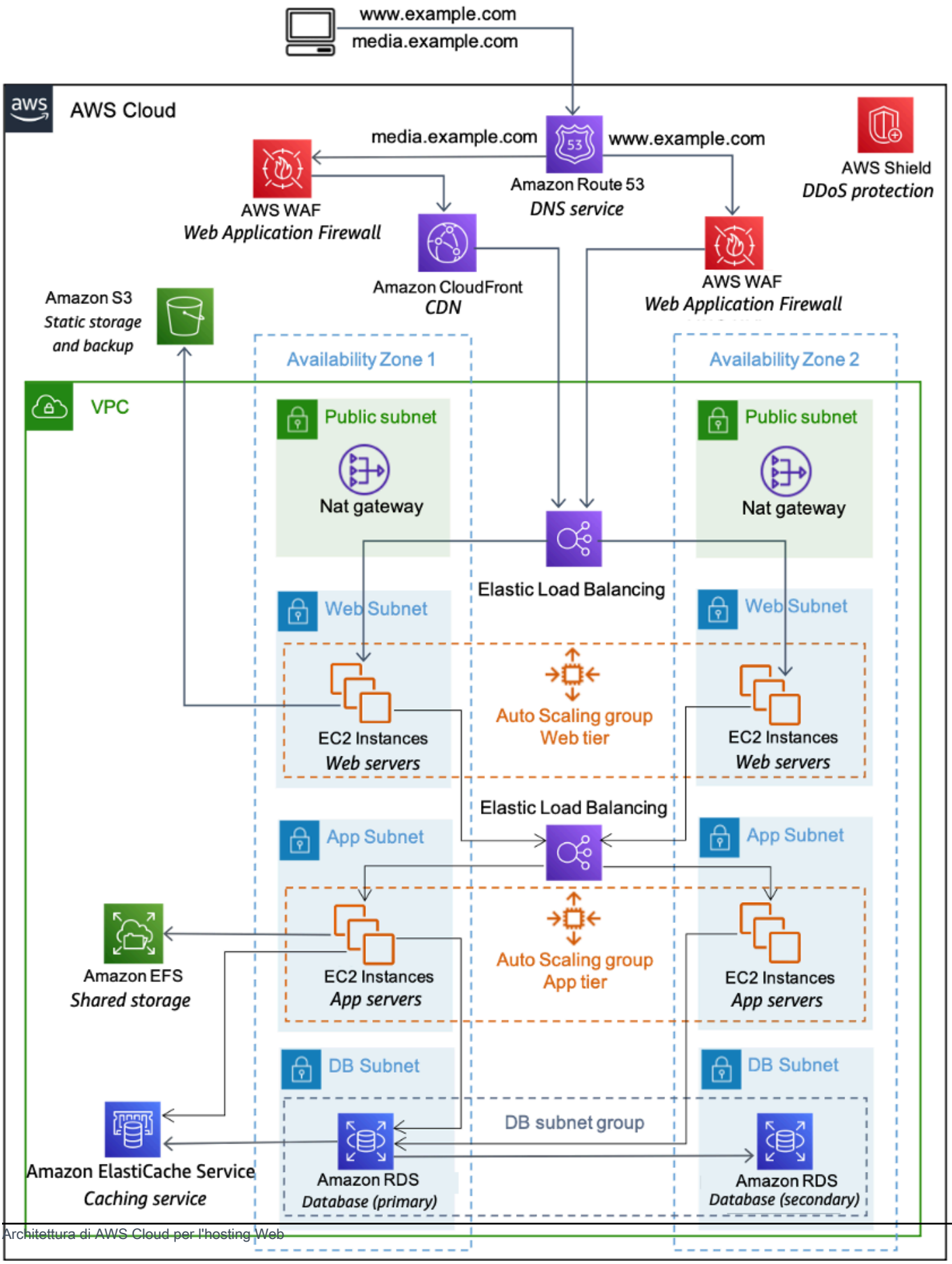
Soluzione on demand per ambienti di test, carico, beta e riproduzione

I costi hardware per la creazione e la manutenzione di un ambiente di hosting tradizionale per un'applicazione Web di produzione non si fermano al parco istanze di produzione. Spesso è necessario creare parchi istanze di pre-produzione, beta e test per verificare la qualità dell'applicazione Web in ogni fase del ciclo di vita di sviluppo. Sebbene sia possibile apportare varie ottimizzazioni per usare al meglio l'hardware di test, questi parchi istanze paralleli non vengono sempre usati in modo ottimale e molto hardware costoso rimane inutilizzato per lunghi periodi di tempo.

In AWS Cloud è possibile effettuare il provisioning di parchi istanze di test come e quando è necessario. In questo modo, non solo si elimina la necessità di pre-provisioning delle risorse giorni o mesi prima dell'utilizzo effettivo, ma si ottiene anche la flessibilità necessaria per rimuovere i componenti dell'infrastruttura quando non sono necessari. È inoltre possibile simulare il traffico degli utenti in AWS Cloud durante i test di carico. È anche possibile usare questi parchi istanze paralleli come ambiente di gestione temporanea per una nuova versione di produzione. Sarà così possibile passare rapidamente dalla versione di produzione corrente a una nuova versione dell'applicazione senza interruzioni del servizio o con interruzioni minime.

Architettura di AWS Cloud per l'hosting Web

La figura seguente mostra un altro esempio di architettura di un'applicazione Web classica e di come possa usufruire dell'infrastruttura di cloud computing AWS Cloud.


www.example.com
media.example.com
AWS Cloud
media.example.com
www.example.com
AWS Shield
DDoS protection
AWS WAF
Web Application Firewall
Amazon Route 53
DNS service
Amazon CloudFront
CDN
AWS WAF
Web Application Firewall
Amazon S3
Static storage and backup
Availability Zone 1
Availability Zone 2
VPC
Public subnet
Nat gateway
Public subnet
Nat gateway
Elastic Load Balancing
Web Subnet
Web Subnet
EC2 Instances
Web servers
Auto Scaling group
Web tier
EC2 Instances
Web servers
Elastic Load Balancing
App Subnet
App Subnet
Amazon EFS
Shared storage
EC2 Instances
App servers
Auto Scaling group
App tier
EC2 Instances
App servers
DB Subnet
DB Subnet
DB subnet group
Amazon ElastiCache Service
Caching service
Amazon RDS
Database (primary)
Amazon RDS
Database (secondary)

Esempio di architettura di hosting Web in AWS

1. Servizi DNS con Amazon Route 53: servizi DNS per semplificare la gestione del dominio.
2. Memorizzazione nella cache edge con Amazon CloudFront: memorizzazione nella cache edge di grandi volumi di contenuti per ridurre la latenza per i clienti.
3. Sicurezza edge per Amazon CloudFront con AWS WAF: filtro del traffico dannoso, tra cui cross-site scripting (XSS) e SQL injection, tramite regole definite dal cliente.
4. Bilanciamento del carico con Elastic Load Balancing (ELB): distribuzione del carico tra più zone di disponibilità e gruppi AWS Auto Scaling per la ridondanza e il disaccoppiamento dei servizi.
5. Protezione DDoS con AWS Shield: protezione automatica dell'infrastruttura dagli attacchi DDoS più comuni a livello di rete e trasporto.
6. Firewall con gruppi di sicurezza: spostamento della sicurezza sull'istanza per fornire un firewall con stato a livello di host sia per i server Web che per i server applicazioni.
7. Memorizzazione nella cache con Amazon ElastiCache: servizi di memorizzazione nella cache con Redis o Memcached per rimuovere il carico dall'app e dal database e ridurre la latenza per le richieste frequenti.
8. Database gestito con Amazon Relational Database Service (Amazon RDS): creazione di un'architettura di database Multi-AZ a disponibilità elevata con sei possibili motori di database.
9. Backup e archiviazione di tipo statico con Amazon Simple Storage Service (Amazon S3): semplice archiviazione di oggetti basata su HTTP per backup e risorse statiche come immagini e video.

Componenti chiave di un'architettura di hosting Web AWS

Le sezioni seguenti illustrano alcuni dei componenti chiave di un'architettura di hosting Web distribuita in AWS Cloud, illustrandone le differenze da un'architettura di hosting Web tradizionale.

Gestione delle reti

In AWS Cloud la possibilità di segmentare la rete rispetto a quella di altri clienti offre un'architettura più sicura e scalabile. Mentre i gruppi di sicurezza forniscono sicurezza a livello di host (vedi la sezione Sicurezza dell'host), Amazon Virtual Private Cloud (Amazon VPC) permette di avviare le risorse in una rete virtuale e isolata dal punto di vista logico, che è possibile definire.

Amazon VPC è un servizio che offre il controllo completo sui dettagli della configurazione di rete in AWS. È ad esempio possibile creare sottoreti pubbliche per i server Web e sottoreti private senza accesso a Internet per i database. Amazon VPC ti permette inoltre di creare architetture ibride

usando reti private virtuali (VPN) hardware, nonché di usare AWS Cloud come estensione del tuo data center.

Amazon VPC include inoltre il supporto per IPv6 oltre al tradizionale supporto di IPv4 per la rete.

Distribuzione di contenuti

Quando il traffico Web è distribuito in aree geografiche diverse, non è sempre fattibile né conveniente replicare l'intera infrastruttura in tutto il mondo. Una rete per la distribuzione di contenuti (CDN, Content Delivery Network) offre la possibilità di impiegare la rete globale di posizioni edge per offrire ai clienti una copia cache di contenuti Web tra cui video, pagine Web, immagini e così via. Per ridurre il tempo di risposta, la rete CDN utilizza la posizione edge più vicina al cliente o la posizione da cui ha avuto origine la richiesta. La velocità effettiva risulta notevolmente migliore, perché le risorse Web provengono dalla cache. Per quanto riguarda i dati dinamici, molte reti CDN possono essere configurate per recuperare i dati dai server di origine.

È possibile usare CloudFront per distribuire il website, inclusi contenuti dinamici, statici e in streaming, usando una rete globale di posizioni edge. CloudFront instrada automaticamente le richieste di contenuti alla posizione edge più vicina, in modo da offrire il miglior livello possibile di prestazioni per la distribuzione dei contenuti. CloudFront è ottimizzato per il funzionamento con altri servizi AWS come Amazon S3 e Amazon Elastic Compute Cloud (Amazon EC2). CloudFront funziona inoltre in modo ottimale con qualsiasi server di origine non AWS in cui sono archiviate le versioni originali e definitive dei file.

Come altri servizi AWS, non sono previsti contratti o impegni mensili per l'uso di CloudFront: paghi solo in base alla quantità di contenuti effettivamente distribuiti attraverso il servizio.

Inoltre, tutte le soluzioni esistenti per la memorizzazione nella cache edge nell'infrastruttura delle applicazioni Web dovrebbero funzionare bene in AWS Cloud.

Gestione dei nomi DNS pubblici

Lo spostamento di un'applicazione Web in AWS Cloud richiede alcune modifiche al sistema Domain Name System (DNS). Per offrire supporto per la gestione del routing DNS, AWS fornisce Amazon Route 53, un servizio Web DNS cloud con disponibilità e scalabilità elevate. Route 53 è un servizio concepito per fornire a sviluppatori e aziende un modo estremamente affidabile e conveniente per instradare gli utenti finali alle applicazioni Internet convertendo nomi come "www.esempio.com" in indirizzi IP numerici come 192.0.2.1, che i computer usano per connettersi tra loro. Route 53 è inoltre completamente conforme a IPv6.

Sicurezza dell'host

Oltre a filtrare il traffico di rete in entrata nell'ambiente edge, AWS consiglia anche che le applicazioni Web applichino filtri del traffico di rete a livello di host. Amazon EC2 offre una caratteristica denominata gruppi di sicurezza. Un gruppo di sicurezza è analogo a un firewall di rete in entrata, per il quale è possibile specificare i protocolli, le porte e gli intervalli IP di origine autorizzati a raggiungere le istanze EC2.

È possibile assegnare uno o più gruppi di sicurezza a ogni istanza EC2. Ogni gruppo di sicurezza permette il flusso del traffico appropriato verso ogni istanza. È possibile configurare i gruppi di sicurezza in modo che solo sottoreti, risorse e indirizzi IP specifici abbiano accesso a un'istanza EC2. In alternativa, è possibile fare riferimento ad altri gruppi di sicurezza per limitare l'accesso alle istanze EC2 che si trovano in gruppi specifici.

Nell'architettura di hosting Web AWS nella Figura 3, il gruppo di sicurezza per il cluster di server Web può permettere l'accesso solo dal bilanciatore del carico a livello Web e solo tramite TCP sulle porte 80 e 443 (HTTP e HTTPS). Il gruppo di sicurezza del server applicazioni, d'altra parte, può permettere l'accesso solo dal bilanciatore del carico a livello di applicazione. In questo modello, i tecnici del supporto necessitano anche di accesso alle istanze EC2, che è possibile ottenere con AWS Systems Manager Session Manager. Per una discussione più approfondita sulla sicurezza, consulta Sicurezza in AWS Cloud, che contiene bollettini sulla sicurezza, informazioni sulle certificazioni e whitepaper sulla sicurezza che spiegano le funzionalità di sicurezza di AWS.

Bilanciamento del carico tra cluster

I bilanciatori del carico hardware sono appliance di rete comuni in uso nelle architetture delle applicazioni Web tradizionali. AWS fornisce questa funzionalità tramite il servizio Elastic Load Balancing (ELB). ELB distribuisce automaticamente il traffico in entrata dell'applicazione tra più destinazioni, come istanze EC2, container, indirizzi IP, funzioni AWS Lambda e applicazioni virtuali. Permette di gestire i diversi carichi di traffico di un'applicazione in una o più zone di disponibilità. Elastic Load Balancing offre quattro tipi di bilanciatori del carico, tutti dotati di disponibilità elevata, scalabilità automatica e sicurezza affidabile, elementi necessari per permettere la tolleranza ai guasti delle applicazioni.

Ricerca di altri host e servizi

Nella tradizionale architettura di hosting Web, la maggior parte degli host ha indirizzi IP statici. In AWS Cloud, la maggior parte degli host ha indirizzi IP dinamici. Sebbene ogni istanza EC2 possa

avere voci DNS sia pubbliche che private e possa essere indirizzabile in Internet, le voci DNS e gli indirizzi IP vengono assegnati dinamicamente all'avvio dell'istanza. Non possono essere assegnati manualmente. Gli indirizzi IP statici (indirizzi IP elastici nella terminologia AWS) possono essere assegnati alle istanze in esecuzione dopo l'avvio. È consigliabile usare indirizzi IP elastici per le istanze e i servizi che richiedono endpoint coerenti, come database primari, file server centrali e bilanciatori del carico ospitati da EC2.

Memorizzazione nella cache all'interno dell'applicazione Web

Le cache delle applicazioni in memoria permettono di ridurre il carico sui servizi e migliorare le prestazioni e la scalabilità a livello di database memorizzando le informazioni usate di frequente. Amazon ElastiCache è un servizio Web che permette di distribuire, gestire e dimensionare una cache in memoria nel cloud con la massima semplicità. È possibile configurare la cache in memoria creata per ottenere scalabilità automatica in base al carico e sostituire automaticamente i nodi in cui si verificano errori. ElastiCache è conforme ai protocolli Memcached e Redis, il che semplifica la migrazione dalla soluzione On-Premise corrente.

Configurazione, backup e failover del database

Molte applicazioni Web contengono una forma di persistenza, di solito sotto forma di database relazionale o non relazionale. AWS offre servizi di database relazionali e non relazionali. In alternativa, è possibile distribuire il proprio software di database in un'istanza EC2. La tabella seguente riepiloga queste opzioni, che vengono discusse più dettagliatamente in questa sezione.

Tabella 1. Soluzioni di database relazionali e non relazionali

	Soluzioni di database relazionali	Soluzioni NoSQL
Servizio di database gestito	Amazon RDS for MySQL, Oracle, SQL Server, MariaDB, PostgreSQL, Amazon Aurora	Amazon DynamoDB, Amazon Keyspaces, Amazon Neptune, Amazon QLDB, Amazon Timestream
Autogestito	Hosting di un sistema di gestione di database relazionale (DBMS) in un'istanza Amazon EC2	Hosting di una soluzione di database non relazionale in un'istanza EC2

Amazon RDS

Amazon Relational Database Service (Amazon RDS) permette di accedere alle funzionalità di un familiare motore di database MySQL, PostgreSQL, Oracle e Microsoft SQL Server. Con Amazon RDS puoi usare il codice, le applicazioni e gli strumenti che già conosci. Amazon RDS applica automaticamente patch al software di database ed esegue i backup del database, archiviandoli per un periodo di conservazione definito dall'utente. Supporta anche il ripristino point-in-time (PITR). Hai così la flessibilità per dimensionare le risorse di calcolo o la capacità di archiviazione per l'istanza del database relazionale attraverso una singola chiamata API.

Le implementazioni Multi-AZ di Amazon RDS aumentano la disponibilità del database e proteggono il database da interruzioni non pianificate. Le repliche di lettura di Amazon RDS forniscono repliche di sola lettura del database, così da poter aumentare orizzontalmente la capacità rispetto alla distribuzione di un singolo database per i carichi di lavoro di database gravosi in lettura. Come per tutti i servizi AWS, non sono richiesti investimenti iniziali e si paga solo per le risorse usate.

Hosting di un sistema di gestione di database relazionale (RDBMS) in un'istanza Amazon EC2

Oltre all'offerta gestita di Amazon RDS, è possibile installare il sistema RDBMS desiderato (ad esempio MySQL, Oracle, SQL Server o DB2) in un'istanza EC2 e gestirlo personalmente. I clienti AWS che ospitano un database in Amazon EC2 possono usare diversi modelli di replica e primario/di standby, tra cui il mirroring per copie di sola lettura e il log shipping per slave passivi sempre disponibili.

Quando gestisci il software di database direttamente in Amazon EC2, devi considerare anche la disponibilità di risorse di archiviazione persistente e con tolleranza ai guasti. A tale scopo, è consigliabile che i database in esecuzione in Amazon EC2 usino volumi Amazon Elastic Block Store (Amazon EBS), analoghi a dispositivi NAS (Network-Attached Storage).

Per le istanze EC2 che eseguono un database, è consigliabile inserire tutti i dati e i registri del database nei volumi EBS. Questi rimarranno disponibili anche in caso di errore dell'host del database. Questa configurazione permette un semplice scenario di failover, in cui è possibile avviare una nuova istanza EC2 in caso di errore di un host e i volumi EBS esistenti possono essere collegati alla nuova istanza. Il database può quindi riprendere da dove si è interrotto.

I volumi EBS forniscono automaticamente ridondanza all'interno della zona di disponibilità. Se le prestazioni di un singolo volume EBS non sono sufficienti per le esigenze del database, i volumi

possono essere sottoposti a striping per aumentare le prestazioni in termini di operazioni di input/output al secondo (IOPS) per il database.

Per carichi di lavoro impegnativi è anche possibile usare volumi EBS IOPS con provisioning, specificando il livello di IOPS richiesto. Se usi Amazon RDS, il servizio gestisce le risorse di archiviazione permettendoti così di concentrarti sulla gestione dei dati.

Database non relazionali

Oltre al supporto per i database relazionali, AWS offre anche diversi database non relazionali gestiti:

- Amazon DynamoDB è un servizio di database NoSQL completamente gestito che combina prestazioni elevate e prevedibili con una scalabilità ottimale. Usando la AWS Management Console o l'API DynamoDB, è possibile dimensionare la capacità senza tempo di inattività né riduzione delle prestazioni. Poiché DynamoDB si occupa delle attività amministrative connesse alla gestione e al dimensionamento dei database distribuiti in AWS, non dovrai occuparti di provisioning dell'hardware, installazione e configurazione, replica, applicazione di patch software o dimensionamento dei cluster.
- Amazon DocumentDB (con compatibilità con MongoDB) è un servizio di database creato appositamente per la gestione di dati JSON su larga scala, completamente gestito, eseguito in AWS e pronto per le aziende con elevata durabilità.
- Amazon Keyspaces (per Apache Cassandra) è un servizio di database compatibile con Apache Cassandra, gestito, scalabile e a disponibilità elevata. Con Amazon Keyspaces è possibile eseguire i carichi di lavoro Cassandra in AWS usando lo stesso codice applicativo e gli stessi strumenti di sviluppo Cassandra già in uso.
- Amazon Neptune è un servizio di database a grafo rapido, affidabile e completamente gestito che semplifica la costruzione e l'esecuzione di applicazioni che funzionano con set di dati altamente connessi. Il centro nevralgico di Amazon Neptune è un motore di database a grafo dedicato ad alte prestazioni, ottimizzato per archiviare miliardi di relazioni ed eseguire query sul grafo con una latenza di millisecondi.
- Amazon Quantum Ledger Database (Amazon QLDB) (QLDB) è un database di libro mastro completamente gestito che fornisce un registro delle transazioni trasparente, immutabile e verificabile crittograficamente di proprietà di un'autorità attendibile centrale. QLDB permette di monitorare ogni cambiamento dei dati di un'applicazione e conserva una cronologia completa e verificabile delle modifiche nel corso del tempo.
- Amazon Timestream è un servizio di database di serie temporali rapido, scalabile e serverless pensato per applicazioni IoT e operative. Questo servizio semplifica l'archiviazione e l'analisi di

trilioni di eventi al giorno fino a 1.000 volte più rapidamente e a un decimo del costo rispetto ai database relazionali.

È inoltre possibile usare Amazon EC2 per ospitare altre tecnologie di database non relazionali.

Archiviazione e backup di dati e risorse

AWS Cloud offre numerose opzioni per accedere alle risorse e ai dati delle applicazioni Web, archiviarli ed eseguirne il backup. Amazon S3 offre un archivio oggetti ridondante e a disponibilità elevata. Amazon S3 è un'ottima soluzione di archiviazione per oggetti statici o che cambiano lentamente, come immagini, video e altri contenuti multimediali statici. Amazon S3 supporta anche la memorizzazione nella cache edge e lo streaming di queste risorse tramite l'interazione con CloudFront.

Per l'archiviazione simile a quella di un file system collegato, è possibile collegare volumi EBS alle istanze EC2. Tali volumi si comportano come dischi montabili per l'esecuzione di istanze EC2. Amazon EBS è ideale per i dati a cui è necessario accedere come archiviazione a blocchi e che richiedono persistenza oltre la durata dell'istanza in esecuzione, ad esempio partizioni di database e registri delle applicazioni.

Oltre ad avere una durata indipendente dall'istanza EC2, i volumi EBS permettono di acquisire snapshot e archiviarli in Amazon S3. Poiché gli snapshot EBS eseguono solo il backup delle modifiche rispetto allo snapshot precedente, l'acquisizione più frequente riduce i tempi di creazione degli snapshot. È inoltre possibile usare uno snapshot EBS come base per la replica dei dati tra più volumi EBS e il collegamento di tali volumi ad altre istanze in esecuzione.

I volumi EBS possono avere dimensioni fino a 16 TB ed è possibile sottoporre più volumi EBS a striping per ottenere dimensioni ancora più grandi o per migliorare le prestazioni di input/output (I/O). Per ottimizzare le prestazioni delle applicazioni con numerose operazioni di I/O, è possibile usare i volumi IOPS con provisioning. I volumi IOPS con provisioning sono progettati per soddisfare le esigenze dei carichi di lavoro onerosi in termini di I/O, in particolare i carichi di lavoro di database sensibili alle prestazioni delle risorse di archiviazione e alla coerenza della velocità effettiva di I/O ad accesso casuale.

L'utente specifica una frequenza IOPS al momento della creazione del volume. e Amazon EBS ne effettua il provisioning per la durata del volume. Amazon EBS attualmente supporta un numero di IOPS per volume che va da un massimo di 16.000 (per tutti i tipi di istanza) fino a 64.000 (per le istanze basate su Nitro System). È possibile eseguire lo striping di più volumi per distribuire migliaia di IOPS per istanza per l'applicazione. Oltre a questo, per una velocità effettiva più elevata e i carichi

di lavoro mission-critical che richiedono una latenza inferiore al millisecondo, è possibile usare il tipo di volume io2 block express in grado di supportare fino a 256.000 IOPS con una capacità di archiviazione massima di 64 TB.

Scalabilità automatica del parco istanze

Una delle principali differenze tra l'architettura di AWS Cloud e il modello di hosting tradizionale è che AWS offre scalabilità automatica del parco istanze dell'applicazione Web on demand per far fronte ai cambiamenti del traffico. Con il modello di hosting tradizionale, vengono in genere usati modelli di previsione del traffico per effettuare il provisioning degli host in anticipo rispetto al traffico previsto. In AWS è possibile effettuare il provisioning delle istanze in tempo reale in base a un set di meccanismi di attivazione tramite cui dimensionare il parco istanze aumentando o diminuendo la capacità.

Il servizio Auto Scaling permette di creare gruppi di capacità di server che è possibile aumentare o ridurre on demand. Auto Scaling funziona anche direttamente con CloudWatch per i dati dei parametri e con Elastic Load Balancing per l'aggiunta e la rimozione di host per la distribuzione del carico. Se, ad esempio, i server Web segnalano un utilizzo della CPU superiore all'80% in un determinato periodo di tempo, è possibile implementare rapidamente un server Web aggiuntivo e quindi aggiungerlo automaticamente al bilanciatore del carico in modo da inserirlo immediatamente nella rotazione di bilanciamento del carico.

Come illustrato nel modello di architettura di hosting Web AWS, è possibile creare più gruppi Auto Scaling per diversi livelli dell'architettura, in modo che ogni livello possa essere dimensionato in modo indipendente. Il gruppo Auto Scaling del server Web può ad esempio attivare il dimensionamento in risposta alle modifiche nell'I/O di rete, mentre il gruppo Auto Scaling del server applicazioni può aumentare orizzontalmente la capacità in base all'utilizzo della CPU. È possibile impostare valori minimi e massimi per assicurare la disponibilità 24 ore su 24, 7 giorni su 7 e per limitare l'utilizzo all'interno di un gruppo.

I meccanismi di attivazione di Auto Scaling possono essere impostati sia per aumentare che per ridurre il parco istanze totale in un determinato livello in modo da far corrispondere l'utilizzo delle risorse alla domanda effettiva. Oltre al servizio Auto Scaling, puoi dimensionare i parchi istanze Amazon EC2 direttamente tramite l'API di Amazon EC2, che permette di avviare, terminare e ispezionare le istanze.

Caratteristiche di sicurezza aggiuntive

Gli attacchi DDoS (Distributed Denial of Service) diventano sempre più frequenti e più sofisticati. Tradizionalmente, questi attacchi sono difficili da respingere. Spesso finiscono per essere costosi sia

in termini di tempo di mitigazione che di energia spesa, nonché di perdita di opportunità derivante dalle visite mancate nel website durante l'attacco. Ci sono numerosi fattori e servizi AWS che possono essere utili per difendersi da tali attacchi. Uno di questi è la scalabilità della rete AWS. L'infrastruttura AWS è piuttosto ampia ed è possibile usufruire di questa scalabilità per ottimizzare la difesa. Diversi servizi, tra cui Elastic Load Balancing, Amazon CloudFront e Amazon Route 53, sono efficaci per dimensionare l'applicazione Web in risposta a un forte aumento del traffico.

I servizi di protezione dell'infrastruttura, in particolare, supportano la strategia di difesa:

- AWS Shield è un servizio di protezione DDoS gestito che aiuta a proteggersi da varie forme di vettori di attacco DDoS. L'offerta standard di AWS Shield è gratuita e attiva automaticamente nell'account. Questa offerta standard aiuta a difendersi dagli attacchi più comuni a livello di rete e di trasporto. Oltre a questo livello, l'offerta avanzata fornisce livelli più elevati di protezione per l'applicazione Web grazie a una visibilità in tempo quasi reale su un attacco in corso e a una maggiore integrazione con i servizi menzionati in precedenza. È inoltre possibile accedere al team DRT (DDoS Response Team) AWS per ottenere supporto per la mitigazione di attacchi sofisticati e su larga scala contro le risorse.
- AWS WAF (Web Application Firewall) è un servizio progettato per proteggere le applicazioni Web da attacchi che possono compromettere la disponibilità o la sicurezza oppure consumare in altro modo una quantità eccessiva di risorse. AWS WAF funziona in linea con CloudFront o Application Load Balancer, insieme alle regole personalizzate, per fornire difesa da attacchi come cross-site scripting (XSS), SQL injection e DDoS. Come per la maggior parte dei servizi AWS, AWS WAF include un'API completa che aiuta ad automatizzare la creazione e la modifica delle regole per l'istanza di AWS WAF man mano che le esigenze di sicurezza cambiano.
- AWS Firewall Manager è un servizio per la gestione della sicurezza che permette di configurare e gestire a livello centrale le regole del firewall per tutti gli account e le applicazioni in AWS Organizations. Quando vengono create nuove applicazioni, AWS Firewall Manager permette di rendere facilmente conformi le nuove applicazioni e risorse implementando un set condiviso di regole di sicurezza.

Failover con AWS

Un altro vantaggio chiave di AWS rispetto all'hosting Web tradizionale è rappresentato dalle zone di disponibilità che offrono un facile accesso a posizioni di implementazione ridondanti. Le zone di disponibilità sono posizioni fisicamente distinte progettate per rimanere isolate dai guasti che si verificano in altre zone di disponibilità. Offrono connettività di rete non costosa e a bassa latenza ad altre zone di disponibilità nella stessa regione AWS. Come mostra il diagramma dell'architettura di

hosting Web AWS, AWS consiglia di implementare host EC2 in più zone di disponibilità per migliorare la tolleranza ai guasti dell'applicazione Web.

È importante assicurarsi che siano previste disposizioni per la migrazione di singoli punti di accesso tra le zone di disponibilità in caso di guasto. È ad esempio necessario configurare lo standby del database in una seconda zona di disponibilità in modo che la persistenza dei dati rimanga coerente e con disponibilità elevata, anche durante un improbabile scenario di guasto. A tale scopo, è sufficiente un clic in Amazon EC2 o Amazon RDS.

Sebbene siano spesso necessarie alcune modifiche dell'architettura quando si sposta un'applicazione Web esistente in AWS Cloud, si ottengono miglioramenti significativi in termini di scalabilità, affidabilità e convenienza che giustificano ampiamente il lavoro necessario per il passaggio ad AWS Cloud. Nella sezione successiva vengono illustrati questi miglioramenti.

Considerazioni chiave per l'uso di AWS per l'hosting Web

Ci sono alcune differenze fondamentali tra AWS Cloud e un modello di hosting di applicazioni Web tradizionale. Nella sezione precedente sono state illustrate molte delle aree chiave da considerare quando si distribuisce un'applicazione Web nel cloud. Questa sezione illustra alcuni dei principali cambiamenti nell'architettura che è necessario considerare quando si sposta un'applicazione nel cloud.

Nessuna appliance di rete fisica

Non è possibile distribuire appliance di rete fisiche in AWS. Ad esempio, firewall, router e bilanciatori del carico per le applicazioni AWS non possono più trovarsi nei dispositivi fisici, ma devono essere sostituiti con soluzioni software. Sono disponibili molte soluzioni software di livello aziendale, sia per il bilanciamento del carico che per la creazione di una connessione VPN. Questa non è una limitazione a ciò che è possibile eseguire in AWS Cloud, ma rappresenta solo una modifica dell'architettura dell'applicazione se attualmente usi questi dispositivi.

Firewall ovunque

Dove un tempo era presente una semplice rete perimetrale (DMZ) con comunicazioni aperte tra gli host in un modello di hosting tradizionale, AWS applica un modello più sicuro, in cui ogni host è bloccato. Uno dei passaggi nella pianificazione di una distribuzione AWS è l'analisi del traffico tra gli host. Questa analisi permetterà di stabilire esattamente quali porte è necessario aprire. È possibile creare gruppi di sicurezza per ogni tipo di host nell'architettura. È anche possibile creare una vasta gamma di modelli di sicurezza semplici e a più livelli per consentire l'accesso minimo tra gli host all'interno dell'architettura. L'uso di liste di controllo accessi di rete in Amazon VPC può aiutare a bloccare la rete a livello di sottorete.

Considerazioni sulla disponibilità di più data center

Le zone di disponibilità in una regione AWS possono essere paragonate a data center multipli. Le istanze EC2 in zone di disponibilità diverse sono separate sia logicamente che fisicamente e forniscono un modello facile da usare per la distribuzione dell'applicazione tra diversi data center, per disponibilità e affidabilità elevate. Amazon VPC è un servizio regionale che permette di usufruire delle zone di disponibilità mantenendo tutte le risorse nella stessa rete logica.

Host effimeri e dinamici

Probabilmente il cambiamento più importante nella progettazione dell'applicazione AWS riguarda il fatto che gli host Amazon EC2 devono essere considerati effimeri e dinamici. Qualsiasi applicazione costruita per AWS Cloud non deve presumere che un host sia sempre disponibile e deve essere progettata con la consapevolezza che tutti i dati negli archivi delle istanze EC2 andranno persi in caso di errore di un'istanza EC2.

Quando viene attivato un nuovo host, non si devono fare supposizioni sull'indirizzo IP o sulla posizione dell'host all'interno di una zona di disponibilità. Il modello di configurazione deve essere flessibile e l'approccio al processo di bootstrap di un host deve tenere conto della natura dinamica del cloud. Queste tecniche sono fondamentali per la costruzione e l'esecuzione di un'applicazione con scalabilità elevata e tolleranza ai guasti.

Considerazioni su container ed elaborazione serverless

Questo whitepaper è incentrato principalmente su un'architettura web più tradizionale. Considera tuttavia la possibilità di modernizzare le applicazioni Web passando a container e tecnologie serverless, usando servizi come AWS Fargate e AWS Lambda per astrarre l'uso di macchine virtuali per l'esecuzione di attività di calcolo. Grazie all'elaborazione serverless, le attività di gestione dell'infrastruttura come il provisioning della capacità e l'applicazione di patch sono gestite da AWS, permettendoti così di costruire applicazioni più agili per innovare e rispondere ai cambiamenti più velocemente.

Considerazioni sulla distribuzione automatica

- Amazon Lightsail è un server privato virtuale di semplice utilizzo che offre tutto il necessario per costruire un'applicazione o un website con un piano mensile conveniente. Lightsail è ideale per i carichi di lavoro più semplici, per le distribuzioni veloci e per iniziare a lavorare con AWS. È un servizio progettato per iniziare in piccolo con la possibilità di dimensionare le risorse man mano che si cresce.
- AWS Elastic Beanstalk è un servizio di facile utilizzo per distribuire e dimensionare applicazioni e servizi Web sviluppati con Java, .NET, PHP, Node.js, Python, Ruby, Go e Docker su server comuni come Apache, NGINX, Passenger e IIS. Caricando semplicemente il codice, Elastic Beanstalk gestisce automaticamente l'implementazione, il provisioning della capacità, il bilanciamento del carico, la scalabilità automatica e il monitoraggio dell'integrità dell'applicazione. Al contempo,

l'utente mantiene il controllo completo sulle risorse AWS su cui si basa la sua applicazione e può accedere in qualsiasi momento alle risorse sottostanti.

- AWS App Runner è un servizio completamente gestito che facilita agli sviluppatori l'implementazione rapida di API e applicazioni Web in container, su larga scala e senza necessità di esperienza precedente relativa all'infrastruttura. Inizia con il codice sorgente o con l'immagine di container. App Runner crea e distribuisce automaticamente l'applicazione Web e bilancia il carico del traffico mediante la crittografia. App Runner si ricalibra anche automaticamente per soddisfare le esigenze di traffico.
- AWS Amplify è un set di strumenti e servizi che è possibile usare insieme o separatamente per aiutare gli sviluppatori di applicazioni mobili e Web front-end a costruire applicazioni complete e scalabili, con tecnologia AWS. Con Amplify è possibile configurare back-end di app e connettere le app in pochi minuti, distribuire app Web statiche con pochi clic e gestire facilmente i contenuti delle app al di fuori della AWS Management Console.

Conclusione e collaboratori

Conclusione

Quando si valuta la migrazione di un'applicazione Web in AWS Cloud, è necessario prendere in considerazione numerosi aspetti concettuali e dell'architettura. I vantaggi di un'infrastruttura conveniente, estremamente scalabile e con tolleranza ai guasti che cresce insieme all'azienda giustificano senza dubbio il lavoro necessario per la migrazione ad AWS Cloud.

Collaboratori

Hanno contribuito alla stesura di questo documento:

- Amir Khairalomoum, Senior Solutions Architect, AWS
- Dinesh Subramani, Senior Solutions Architect, AWS
- Jack Hemion, Senior Solutions Architect, AWS
- Jatin Joshi, Cloud Support Engineer, AWS
- Jorge Fonseca, Senior Solutions Architect, AWS
- Shinduri K S, Solutions Architect, AWS

Approfondimenti

- Distribuzione di un'applicazione basata su Django in Amazon LightSail
- Distribuzione di un sito Web Drupal a elevata disponibilità in Elastic Beanstalk
- Distribuzione di un'applicazione PHP a elevata disponibilità in Elastic Beanstalk
- Distribuzione di un'applicazione Node.js con DynamoDB in Elastic Beanstalk
- Nozioni di base sulle applicazioni Web Linux in AWS Cloud
- Hosting di un sito Web statico
- Hosting di un sito Web statico tramite Amazon S3
- Tutorial: distribuzione di un'applicazione ASP.NET Core con Elastic Beanstalk
- Tutorial: Come distribuire un'applicazione di esempio .NET utilizzando AWS Elastic Beanstalk

Revisioni del documento

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

update-history-change	update-history-description	update-history-date
Whitepaper aggiornato	Sezioni e diagrammi aggiornati con nuovi servizi, caratteristiche e limiti dei servizi.	20 agosto 2021
Whitepaper aggiornato	Etichetta dell'icona aggiornata per "Memorizzazione nella cache con ElastiCache" nella Figura 3.	29 settembre 2019
Whitepaper aggiornato	Diverse sezioni aggiunte e aggiornate in base ai nuovi servizi. Diagrammi aggiornati per maggiore chiarezza e in base ai nuovi servizi. Aggiunta di VPC come metodo di rete standard in AWS in "Gestione delle reti". Aggiunta di una sezione sulla protezione e sulla mitigazione degli attacchi DDoS in "Caratteristiche di sicurezza aggiuntive". Aggiunta di una piccola sezione sulle architetture serverless per l'hosting Web.	1 luglio 2017
Whitepaper aggiornato	Diverse sezioni aggiornate per una maggiore chiarezza. Diagrammi aggiornati per usare le icone AWS. Aggiunta della sezione "Gestione dei nomi DNS pubblici" per i	1 settembre 2012

	dettagli su Amazon Route 53. Sezione "Ricerca di altri host e servizi" aggiornata per maggiore chiarezza. Sezione "Configurazione, backup e failover del database" aggiornata per maggiore chiarezza e per inserire DynamoDB. Sezione "Archiviazione e backup di dati e risorse" ampliata per illustrare i volumi EBS IOPS con provisioning.	
Pubblicazione iniziale	Whitepaper pubblicato.	1 maggio 2010

Avvisi

Il presente documento è fornito a solo scopo informativo. In esso sono illustrate le attuali offerte di prodotti e le prassi di AWS alla data di pubblicazione del documento, offerte che sono soggette a modifica senza preavviso. È responsabilità dei clienti effettuare una propria valutazione indipendente delle informazioni contenute nel presente documento e dell'uso dei prodotti o dei servizi di AWS, ciascuno dei quali viene fornito "così com'è", senza garanzie di alcun tipo, né esplicite né implicite. Il presente documento non dà origine a garanzie, rappresentazioni, impegni contrattuali, condizioni o assicurazioni da parte di AWS, delle sue società affiliate, dei suoi fornitori o dei licenzianti. 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.