Predisposizione dell'infrastruttura agli eventi

Best practice e linee guida AWS

Luglio 2017

Note

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.

Indice

Sintesi

Questo whitepaper descrive linee guida e best practice per i clienti che dispongono di carichi di lavoro in ambienti di produzione distribuiti su Amazon Web Services (AWS) e che desiderano progettare ed effettuare il provisioning delle loro applicazioni basate su cloud in modo da poter gestire, senza intoppi e in modo dinamico, eventi di dimensionamento pianificati, come il lancio di nuovi prodotti o i picchi di traffico stagionali. Tratteremo i principi generali di progettazione e forniremo specifiche best practice e linee guida riguardo a diverse aree concettuali relative alla pianificazione di eventi di infrastruttura. Descriveremo quindi alcune considerazioni e best practice per la prontezza operativa, nonché le attività post-evento.

Introduzione

La predisposizione dell'infrastruttura agli eventi consiste nella progettazione e nella preparazione di eventi programmati di portata significativa che hanno un impatto sull'azienda. Durante questi eventi, è fondamentale che i servizi Web dell'azienda siano affidabili, reattivi e altamente tolleranti ai guasti, in ogni condizione e per qualsiasi cambiamento nei flussi di traffico. Tali eventi possono includere l'espansione in nuove aree geografiche, il lancio di nuovi prodotti o funzionalità, eventi stagionali o annunci di business o eventi di marketing significativi.

Un evento di infrastruttura non correttamente pianificato può avere ripercussioni negative sulla reputazione dell'attività, sulla continuità delle operazioni o sulle finanze dell'azienda. Durante un evento di infrastruttura, eventuali problemi possono manifestarsi in diverse forme, ad esempio: interruzioni impreviste del servizio, calo delle prestazioni dovuto al carico di lavoro, latenza di rete, limitate capacità di storage, limiti di sistema come la frequenza delle chiamate API, limitate quantità di indirizzi IP disponibili, scarsa comprensione dei comportamenti dei componenti di uno stack di applicazioni a causa di un monitoraggio insufficiente, dipendenze impreviste da un componente o un servizio di terze parti non configurato per il dimensionamento o altre condizioni di errore impreviste.

Per ridurre al minimo il rischio di problemi imprevisti durante un evento importante, le aziende devono investire tempo e risorse per la pianificazione e la preparazione, per addestrare i dipendenti e per formulare e documentare processi appropriati. La quantità di investimenti necessari nella pianificazione di eventi di infrastruttura per una determinata applicazione o set di applicazioni basate su cloud può variare a seconda della complessità del sistema e della sua diffusione globale. Indipendentemente dall'ambito o dalla complessità della presenza di un'azienda su cloud, i principi di progettazione e i suggerimenti di best practice forniti in questo whitepaper sono gli stessi.

Con Amazon Web Services (AWS), le aziende possono ampliare la propria infrastruttura in preparazione di un evento di dimensionamento pianificato in modo dinamico, adattabile e con una tariffazione a consumo. Attraverso una ricca gamma di prodotti e servizi elastici e programmabili di Amazon mette a disposizione dei clienti la stessa infrastruttura altamente sicura, affidabile e

veloce con cui gestisce la propria rete globale, consentendo loro rispondere prontamente a rapidi mutamenti di esigenze della loro attività.

Questo whitepaper fornisce best practice e principi di progettazione che servono da guida nella pianificazione e nell'attuazione degli eventi di infrastruttura; spiega inoltre come utilizzare i servizi AWS per far sì che le applicazioni siano pronte al dimensionamento verticale o orizzontale a seconda delle esigenze dell'azienda.

Pianificare la predisposizione dell'infrastruttura agli eventi

Questa sezione descrive un evento di infrastruttura pianificato e le attività che in genere hanno luogo durante tale evento.

Che cos'è un evento di infrastruttura pianificato?

Un *evento di infrastruttura pianificato* è un evento di business programmato e pianificato durante il quale è fondamentale per l'azienda mantenere un servizio Web altamente reattivo, altamente dimensionabile e tollerante ai guasti. L'evento può essere rendersi necessario per campagne di marketing, novità riguardanti il settore dell'azienda, lanci di prodotti, espansione territoriale o attività simili che comportano traffico aggiuntivo per le applicazioni basate sul Web dell'azienda e l'infrastruttura sottostante.

Cosa accade durante un evento di infrastruttura pianificato?

La principale preoccupazione nella maggior parte degli eventi di infrastruttura pianificati è la possibilità di aggiungere capacità all'infrastruttura Web in modo da poter garantire volumi di traffico più elevati. In un ambiente locale tradizionale dotato di capacità di calcolo, storage e risorse di rete di tipo fisico, il reparto IT dell'azienda deve effettuare il provisioning delle capacità aggiuntive in base alle proprie stime di un picco teorico massimo. Questo comporta il rischio di predisporre capacità insufficienti, con conseguente perdita di opportunità per l'azienda a causa del sovraccarico dei server Web, dei lunghi tempi di risposta e di altri errori di runtime.

Con il cloud AWS, l'infrastruttura è programmabile ed elastica. Questo significa che il provisioning può essere effettuato in modo rapido in risposta alla domanda in tempo reale. Inoltre, ciò significa che l'infrastruttura può essere configurata in modo da rispondere ai parametri di sistema in maniera automatica, intelligente e dinamica, aumentando o diminuendo le risorse come necessario – cluster di server Web, throughput assegnato, capacità di storage, core di elaborazione calcolo disponibili, il numero di shard di streaming e così via.

Inoltre, molti servizi AWS sono completamente gestiti. Questi servizi includono storage, database, analisi, applicazioni e implementazione. Questo significa che i clienti AWS non devono più preoccuparsi di complesse operazioni di configurazione di questi servizi per un evento a traffico elevato. I servizi completamente gestiti AWS sono progettati per offrire scalabilità e alta disponibilità.

Di solito, in preparazione di un evento di infrastruttura pianificato, i clienti AWS conducono una revisione del sistema per valutare l'architettura dell'applicazione e la prontezza operativa, tenendo in considerazione sia la scalabilità che la tolleranza ai guasti. Le stime del traffico vengono valutate e confrontate con le prestazioni dell'attività ordinaria, al fine di determinare i parametri delle capacità attuali e la capacità aggiuntiva necessaria. Tutti i potenziali colli di bottiglia e le dipendenze a monte e a valle da servizi di terze parti vengono identificati e affrontati. Qualora l'evento pianificato includa un'espansione di territorio o del pubblico di riferimento, vengono anche considerati aspetti geografici. L'espansione in altre regioni o zone di disponibilità AWS viene avviata prima dell'evento pianificato. Viene inoltre condotta una revisione delle impostazioni dinamiche del sistema AWS del cliente (Auto Scaling, sistema di bilanciamento del carico, geo-routing, elevata disponibilità, misure di failover) per garantire che queste siano configurate per gestire in modo corretto l'aumento previsto di volume e frequenza delle transazioni. Inoltre, vengono considerate e modificate come necessario le impostazioni statiche, come ad esempio i limiti delle risorse AWS e il percorso dei server di origine della rete di distribuzione dei contenuti (CDN, Content Delivery Network).

Anche i meccanismi di monitoraggio e notifica vengono riesaminati e migliorati, se necessario, per fornire visibilità in tempo reale sugli eventi nel momento in cui si verificano e per l'analisi a posteriori dei dati una volta che l'evento pianificato è stato completato.

Durante l'evento pianificato, i clienti AWS possono anche aprire casi di supporto con AWS qualora abbiano bisogno di assistenza nella risoluzione dei problemi o di supporto in tempo reale, ad esempio quando un server smette di funzionare. I clienti che sottoscrivono il piano di supporto Enterprise di AWS hanno l'ulteriore vantaggio di poter parlare immediatamente con i tecnici del supporto e sollevare casi di gravità critica se occorre una risposta tempestiva.

Dopo l'evento, le risorse di AWS sono progettate per riadeguarsi in modo automatico per soddisfare i livelli di traffico, aumentando o diminuendo a seconda delle necessità degli eventi.

Principi di progettazione

La preparazione per gli eventi pianificati comincia con una buona progettazione prima di implementare qualsiasi stack di applicazioni o carico di lavoro basato sul cloud.

Carichi di lavoro discreti

Una buona progettazione è essenziale per gestire in modo efficace i carichi di lavoro derivanti da eventi pianificati sia a livelli di traffico normali che elevati. A tal fine, assicurarsi sin dall'inizio di progettare raggruppamenti funzionali delle risorse discreti e indipendenti, centrati su specifici prodotti o applicazioni aziendali. Questa sezione descrive le molteplici dimensioni presenti in questo obiettivo progettuale.

Uso dei tag

Per etichettare e organizzare le risorse vengono utilizzati tag. Si tratta di un componente essenziale nella gestione delle risorse infrastrutturali durante un evento di infrastruttura pianificato. Su AWS, i tag consistono in coppie chiave-valore gestite dal cliente che vengono applicate a singole risorse gestite, come il sistema di bilanciamento del carico o un'istanza di Amazon Elastic Compute Cloud (EC2). Facendo riferimento a tag ben definiti collegati a risorse AWS, è possibile identificare facilmente quali risorse all'interno dell'infrastruttura globale partecipano al carico di lavoro durante l'evento programmato. Utilizzando queste informazioni, è possibile quindi analizzarle per stabilire il livello di preparazione. I tag possono anche essere utilizzati per l'allocazione dei costi.

I tag possono essere utilizzati per organizzare le istanze EC2, le immagini Amazon Machine Image (AMI), i sistemi di bilanciamento del carico, i gruppi di sicurezza, le risorse di Amazon Relational Database Service (RDS), le risorse di Amazon Virtual Private Cloud (VPC), i controlli dello stato di Amazon Route 53 e i bucket di Amazon Simple Storage Service (S3), per citare alcuni esempi.

Per ulteriori informazioni sulle strategie più efficaci di utilizzo dei tag, consultare AWS Tagging Strategies.[1]

Per esempi su come creare e gestire i tag e inserirli in gruppi di risorse, consultare Resource Groups and Tagging for AWS.[2]

Legame debole

Quando si realizza un'architettura per il cloud, è necessario progettare i componenti dello stack di applicazioni in modo che possano operare in maniera il più indipendente possibile gli uni dagli altri. Questo conferirà ai carichi di lavoro basati su cloud il vantaggio di resilienza e scalabilità.

È possibile ridurre le interdipendenze tra i componenti in uno stack di applicazioni basato su cloud progettando ogni componente come una black-box, con interfacce di input e output ben definite (per esempio, le API RESTful). Se i componenti non sono applicazioni ma servizi che insieme costituiscono un'applicazione, si parla di *architettura di microservizi*. Per la comunicazione e il coordinamento tra i componenti delle applicazioni, è possibile utilizzare meccanismi di notifica basati su eventi, come ad esempio code di messaggi AWS, per trasferire messaggi tra i componenti, come mostrato nella Figura 1.

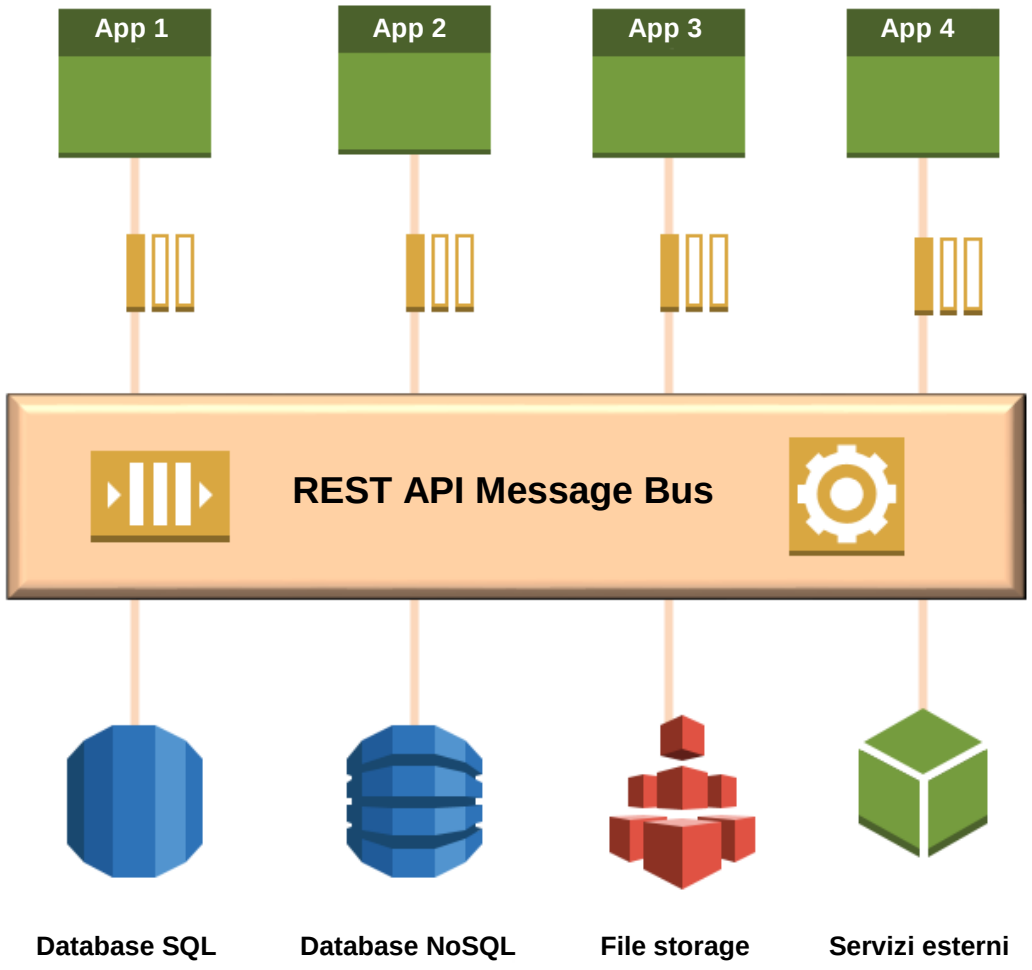

Figura 1. Legame debole mediante interfacce RESTful e code di messaggi

Utilizzando simili meccanismi, una modifica o un errore in un componente ha molte meno probabilità di ripercuotersi su altri componenti. Ad esempio, se un server in uno stack di applicazioni multilivello non risponde, le applicazioni con legame debole possono essere progettate per ignorare il livello che non risponde o passare a transazioni alternative in modalità degradata.

I componenti delle applicazioni con legame debole che utilizzano code di messaggi intermedi possono essere progettati più facilmente per l'integrazione asincrona. Poiché i componenti di un'applicazione non comunicano in modo diretto da punto a punto, ma utilizzano invece un livello di messaggistica intermedio e persistente (ad esempio, una coda Amazon Simple Queue Service (SQS) o un meccanismo di streaming di dati, come Amazon Kinesis Streams), il sistema può sostenere improvvisi aumenti di attività in un componente mentre i

componenti a valle elaborano la coda in entrata. Oppure, se si verifica un errore in un componente, i messaggi persistono nelle code e negli stream finché il componente compromesso non è in grado di recuperare la funzionalità.

Per ulteriori informazioni sulle code di messaggi e sui servizi di notifica offerti da AWS, consultare Amazon Simple Queue Service.[3]

Servizi, non server

I servizi gestiti e gli endpoint di servizio liberano dalle preoccupazioni relative alla sicurezza o all'accesso, ai backup o ai ripristini, alla gestione delle patch o al controllo dei cambiamenti, alla configurazione del monitoraggio o del reporting o alla gestione di dettagli di amministrazione dei sistemi tradizionali. È possibile predisporre queste risorse cloud per ottenere un'elevata disponibilità e resilienza utilizzando configurazioni di zone di disponibilità multiple (o, in alcuni casi, di regioni multiple). Queste risorse possono essere aumentate o ridotte, spesso senza tempi di inattività, e configurate in modo immediato sia tramite la Console di gestione AWS che tramite chiamate API/CLI.

I servizi gestiti e gli endpoint di servizio possono essere utilizzati per introdurre negli stack delle applicazioni del cliente funzionalità aggiuntive, quali database relazionali e NoSQL, data warehousing, notifiche di eventi, archiviazione di oggetti e file, streaming in tempo reale, analisi di Big Data, apprendimento automatico, ricerca, transcoding e molte altre. Un endpoint è un URL che costituisce il punto di accesso ad un servizio AWS. Ad esempio, https://dynamodb.us-west-2.amazonaws.com è un punto di accesso al servizio Amazon DynamoDB.

L'utilizzo di servizi gestiti e dei loro endpoint di servizio permette di sfruttare la potenza di risorse capaci di soddisfare i requisiti di produzione nell'ambito di una soluzione per gestire l'aumento di volume, di portata e di frequenza delle transazioni durante un evento di infrastruttura pianificato. Il cliente non dovrà allestire e gestire server propri che eseguono le stesse funzioni dei servizi gestiti.

Per ulteriori informazioni sugli endpoint dei servizi AWS, consultare AWS Regions and Endpoints[4]. Consultare anche Amazon EMR[5], Amazon RDS[6] e Amazon ECS[7] per esempi di servizi gestiti che dispongono di endpoint.

Architetture serverless

Un'altra strategia per affrontare in modo efficace la necessità di rispondere in maniera dinamica a carichi di elaborazione in continua evoluzione durante un evento di infrastruttura pianificato è utilizzare AWS Lambda. Lambda è una piattaforma di calcolo serverless basata sugli eventi. È un servizio invocato in modo dinamico che esegue codici Python, Node.js o Java in risposta a eventi (tramite notifiche) e gestisce automaticamente le risorse di calcolo indicate da tale codice. Lambda non richiede il pre-provisioning delle risorse di calcolo di Amazon EC2. Il servizio Amazon Simple Notification Service (Amazon SNS) può essere configurato in modo da invocare funzioni Lambda. Per ulteriori informazioni su Amazon SNS, consultare Amazon Push Notification Service.[8]

Le funzioni serverless Lambda sono in grado di eseguire codici che accedono o invocano altri servizi AWS, come operazioni di database, trasformazioni di dati, recupero di oggetti o di file o anche operazioni di dimensionamento in risposta a eventi esterni o a parametri interni di carico di sistema. AWS Lambda, inoltre, è in grado di generare nuove notifiche o eventi propri e persino di avviare altre funzioni Lambda.

AWS Lambda offre la possibilità di esercitare un controllo preciso sulle operazioni di dimensionamento durante un evento di infrastruttura pianificato. Ad esempio, Lambda può essere utilizzato per estendere la funzionalità di Auto Scaling ed eseguire azioni quali, ad esempio, inviare notifiche ai sistemi di terze parti che necessitano anch'essi di dimensionamento, oppure aggiungere ulteriori interfacce di rete alle nuove istanze quando vengono assegnate. Consultare Using AWS Lambda con Auto Scaling Ganci del ciclo di vita[9] per esempi su come usare Lambda per personalizzare le operazioni di dimensionamento.

Per ulteriori informazioni su AWS Lambda, consultare What is AWS Lambda?[10]

Automazione

Auto Scaling

Un componente fondamentale nella pianificazione di eventi di infrastruttura è Auto Scaling. La possibilità di aumentare o ridurre automaticamente la capacità di un'applicazione in base a condizioni predefinite aiuta a mantenere disponibili le applicazioni durante variazioni del flusso e del volume di traffico che si verificano durante un evento di infrastruttura pianificato.

AWS fornisce la funzionalità di Auto Scaling in molte delle proprie risorse, tra cui istanze EC2, capacità di database, container, ecc.

Auto Scaling può essere utilizzato per dimensionare automaticamente in base a criteri specificati raggruppamenti di istanze, come ad esempio un gruppo di server che costituiscono un'applicazione basata su cloud. Auto Scaling può essere utilizzato, inoltre, per mantenere un numero fisso di istanze anche quando una di queste si deteriora. Questo dimensionamento automatico e mantenimento del numero di istanze rappresenta la funzionalità principale del servizio Auto Scaling.

Auto Scaling mantiene il numero di istanze specificato mediante l'esecuzione di controlli di stato periodici sulle istanze del gruppo. Se un'istanza si deteriora, il gruppo ne interrompe l'esecuzione e avvia un'altra istanza per sostituirla.

Le policy di Auto Scaling possono essere utilizzate per aumentare o diminuire automaticamente il numero di istanze EC2 in esecuzione in un gruppo di server in base ai cambiamenti nelle condizioni. Quando la policy di dimensionamento è attiva, il gruppo Auto Scaling modifica la capacità del gruppo richiesta e avvia o interrompe le istanze in base alle esigenze, sia in modo dinamico che in maniera programmatica, se vi è un cambiamento prevedibile nel traffico.

Riavvii e ripristino

Un elemento importante in qualsiasi evento di infrastruttura pianificato è disporre di procedure e di automazione per gestire le istanze o i server compromessi e di essere in grado di ripristinarli o riavviarli in modo immediato.

Le istanze EC2 possono essere configurate per il ripristino automatico quando un controllo dello stato di sistema dell'hardware sottostante ha esito negativo. L'istanza sarà riavviata (su un nuovo hardware, se necessario), ma conserverà i propri ID di istanza, indirizzo IP, indirizzi IP elastici, gli allegati dei volumi Amazon Elastic Block Store (EBS) e altri dettagli di configurazione. Per ulteriori informazioni sul ripristino automatico delle istanze EC2, consultare Auto Recovery of Amazon EC2.[11]

Gestione della configurazione e orchestrazione

L'integrazione di strumenti di gestione della configurazione e di orchestrazione per l'organizzazione dello stato delle singole risorse e l'implementazione dello

stack di applicazioni è una parte integrante di una strategia solida, affidabile e reattiva durante gli eventi di infrastruttura pianificati.

Generalmente, gli strumenti di gestione della configurazione si occupano di effettuare il provisioning e la configurazione delle istanze di server, dei sistemi di bilanciamento del carico, di Auto Scaling, dell'implementazione delle singole applicazioni e del monitoraggio del loro stato. Inoltre, questi strumenti offrono la possibilità di integrarsi con servizi aggiuntivi come database, volumi di storage e livelli di caching.

Gli strumenti di orchestrazione, che si trovano a un livello di astrazione superiore rispetto alla gestione della configurazione, forniscono i mezzi per specificare le relazioni di queste varie risorse, consentendo ai clienti di effettuare il provisioning e di gestire molteplici risorse come un'infrastruttura unificata di applicazioni cloud, senza doversi preoccupare delle dipendenze delle varie risorse.

Poiché questi strumenti definiscono e descrivono le singole risorse, così come le loro relazioni a livello di codice, quest'ultimo può essere sottoposto a controllo versione, facilitando così la possibilità di ritornare a versioni precedenti o aprire nuovi rami di codice per scopi di test o di sviluppo. È inoltre possibile definire strumenti di orchestrazione e configurazioni ottimizzate per un evento di infrastruttura, e successivamente ritornare alla configurazione standard alla conclusione di tale evento.

Amazon Web Services raccomanda l'utilizzo dei seguenti strumenti per ottenere un'implementazione e orchestrazione di hardware come codice:

- **AWS Config con Config Rules** o un Partner AWS Config, per fornire un inventario grafico dettagliato e ricercabile delle risorse AWS, della cronologia di configurazione e della conformità della configurazione delle risorse.
- **AWS CloudFormation** o uno strumento di terze parti per l'orchestrazione delle risorse AWS, per gestire il provisioning, l'aggiornamento e la chiusura delle risorse AWS.
- **AWS OpsWorks, Elastic Beanstalk** o uno strumento di terze parti per la gestione della configurazione dei server, per gestire il sistema operativo e le modifiche alla configurazione delle applicazioni.

Consultare Infrastructure Configuration Management per ulteriori dettagli sui modi per gestire l'hardware come codice.[12]

Diversità e resilienza

Rimozione di singoli punti di errore e colli di bottiglia

Nella pianificazione di un evento di infrastruttura, occorre analizzare gli stack di applicazione per individuare singoli punti di errore (SPOF) o colli di bottiglia delle prestazioni. Esiste, per esempio, un'istanza singola di server, volume di dati, database, gateway NAT o sistema di bilanciamento del carico il cui mancato funzionamento potrebbe causare il blocco della funzionalità dell'applicazione o di una sua parte significativa?

In secondo luogo, quando l'applicazione cloud si ricalibra in termini di traffico o volume di transazioni, esiste una parte dell'infrastruttura che potrebbe incontrare un limite fisico o un vincolo, come ad esempio la larghezza di banda di rete o i cicli di elaborazione di CPU, man mano che il volume di dati cresce all'interno del suo percorso di flusso?

Questi rischi, una volta identificati, possono essere mitigati in diversi modi.

Struttura resiliente ai guasti

Come accennato in precedenza, l'utilizzo del legame debole e delle code di messaggi con interfacce RESTful è una strategia efficace per ottenere la resilienza rispetto ai malfunzionamenti delle singole risorse o delle variazioni del flusso di traffico o del volume delle transazioni. Un altro aspetto della struttura resiliente è quello di configurare i componenti dell'applicazione in modo che siano il più stateless possibile.

Le applicazioni stateless non tengono conto delle transazioni precedenti e hanno una dipendenza debole da altri componenti dell'applicazione. Non mantengono in memoria alcuna informazione sulla sessione. Un'applicazione stateless supporta la scalabilità orizzontale, come membro di un pool o di un cluster, dal momento che ogni richiesta può essere gestita da qualunque istanza all'interno del pool o del cluster. È possibile aggiungere semplicemente più risorse in base alle esigenze utilizzando Auto Scaling e i criteri di controllo dello stato per gestire in modo programmatico le richieste variabili di elaborazione, capacità e throughput. Una volta che un'applicazione è progettata per essere stateless, può potenzialmente essere effettuato il refactoring in un'architettura

serverless, usando le funzioni Lambda al posto delle istanze EC2. Le funzioni Lambda integrano inoltre funzionalità di dimensionamento dinamiche.

Nel caso in cui la risorsa di un'applicazione, ad esempio un server Web, debba mantenere dati di stato relativi alle transazioni, si dovrebbe prendere in considerazione la progettazione dell'applicazione in modo da disaccoppiare le parti stateful dai server stessi. Ad esempio, un cookie HTTP o i dati di stato equivalenti potrebbero essere archiviati in un database, quale DynamoDB, o in un bucket S3 o in un volume EBS.

Se si dispone di un flusso di lavoro complesso a più livelli nel quale è necessario tenere traccia dello stato corrente di ogni fase, è possibile utilizzare il servizio Amazon Simple Workflow (SWF) per archiviare centralmente la cronologia di esecuzione e rendere stateless questi carichi di lavoro.

Un altro criterio di resilienza che è possibile impiegare è l'elaborazione distribuita. Per i casi d'uso che richiedono l'elaborazione di grandi quantità di dati in modo tempestivo, in cui non è sufficiente una singola risorsa di elaborazione, è possibile progettare i carichi di lavoro in modo che le attività e i dati siano suddivisi in porzioni più piccole, ed elaborati in parallelo su un cluster di risorse di elaborazione. L'elaborazione distribuita è stateless, poiché i nodi indipendenti nei quali vengono elaborati i dati e le attività suddivisi potrebbero manifestare dei malfunzionamenti. In questo caso, il motore di pianificazione dell'elaborazione distribuita gestisce automaticamente il riavvio delle attività non riuscite su un nodo diverso del cluster di elaborazione distribuita.

AWS offre un'ampia gamma di motori di elaborazione distribuita dei dati, come Amazon EMR, Amazon Athena e Amazon Machine Learning, ciascuno dei quali è un servizio gestito che fornisce endpoint ed elimina la complessità legata ad applicazione di patch, manutenzione, dimensionamento, failover, ecc.

Per l'elaborazione in tempo reale di flussi di dati, Amazon Kinesis Streams è in grado di suddividere i dati in più shard, che possono essere elaborati da molteplici utilizzatori di tali dati, come funzioni Lambda o istanze EC2.

Per ulteriori informazioni su questi tipi di carichi di lavoro, consultare Big Data Analytics Options on AWS.[13]

Zone multiple e regioni multiple

I servizi AWS sono ospitati in più sedi in tutto il mondo. Queste sedi sono costituite da regioni e zone di disponibilità. Una regione è un'area geografica distinta. Ogni regione dispone di numerose ubicazioni isolate, note come zone di disponibilità. AWS offre ai clienti la possibilità di posizionare le risorse, ad esempio le istanze e i dati, in ubicazioni multiple.

È consigliabile progettare le applicazioni in modo che vengano distribuite in più zone di disponibilità e regioni. In combinazione con la distribuzione e la replica di risorse tra le zone di disponibilità e regioni, è opportuno progettare le applicazioni utilizzando meccanismi di bilanciamento del carico e di failover, in modo che il flusso di dati e il traffico dello stack di applicazioni vengano automaticamente reinstradati verso queste ubicazioni alternative in caso di errore.

Bilanciamento del carico

Con il servizio di Elastic Load Balancing (ELB), un gruppo di server di applicazioni può essere associato a un sistema di bilanciamento del carico e tuttavia essere distribuito su più zone di disponibilità. Quando le istanze EC2 di una determinata zona di disponibilità associate a un sistema di bilanciamento del carico non superano i controlli di stato, il sistema di bilanciamento del carico smette di inviare il traffico verso questi nodi. Se usato in combinazione con Auto Scaling, il numero di nodi integri viene automaticamente ribilanciato con le altre zone di disponibilità senza che sia necessario alcun intervento manuale.

È inoltre possibile bilanciare il carico tra diverse regioni utilizzando Amazon Route 53 e algoritmi di reindirizzamento DNS basati sulla latenza. Consultare Latency Based Routing per maggiori informazioni.[14]

Strategie di riduzione del carico

Il concetto di *riduzione del carico* nelle infrastrutture basate su cloud consiste nel reindirizzamento o nel trasferimento tramite proxy del traffico altrove per allentare la pressione sui sistemi principali. In alcuni casi, la strategia di riduzione del carico può essere un esercizio di determinazione delle priorità, dove si può scegliere di abbandonare certi stream di traffico o ridurre le funzionalità dell'applicazione in modo da alleggerire il carico di elaborazione e permettere almeno di gestire una parte delle richieste in entrata.

Esistono numerose tecniche che possono essere utilizzate per ridurre il carico. Una di queste è il reindirizzamento DNS basato sulla latenza. In alternativa, è possibile utilizzare la memorizzazione nella cache. La memorizzazione nella cache può essere effettuata in prossimità dell'applicazione, utilizzando un layer di caching in memoria come Amazon ElastiCache. In alternativa, è possibile utilizzare un livello di caching più vicino all'utente, utilizzando una rete globale di distribuzione di contenuti, come Amazon CloudFront.

Per ulteriori informazioni su ElastiCache e CloudFront, consultare Getting Started with ElastiCache[15] e Amazon CloudFront CDN.[16]

Ottimizzazione dei costi

Istanze riservate, Spot e on demand

La possibilità di controllare i costi di provisioning delle risorse nel cloud è strettamente legata alla possibilità di effettuare il provisioning delle risorse in modo dinamico nel cloud in base a parametri di sistema e ad altri criteri di controllo delle prestazioni e dello stato. Con Auto Scaling, l'utilizzo delle risorse può essere strettamente collegato alle effettive esigenze di elaborazione e di storage, riducendo al minimo le spese non necessarie e le risorse sottoutilizzate.

Un altro aspetto relativo al controllo dei costi nel cloud è la possibilità di scegliere tra istanze on demand, istanze riservate (IR) oppure istanze Spot. Vi è inoltre la possibilità di prenotare la capacità per DynamoDB.

Con le istanze on demand, si paga soltanto in base alle istanze EC2 utilizzate. Con le istanze on demand, si paga il costo di utilizzo della capacità di elaborazione su base oraria senza alcun impegno a lungo termine.

Le istanze riservate Amazon EC2 offrono notevoli sconti (fino al 75%) rispetto ai prezzi delle istanze on demand; inoltre, se utilizzate in una zona di disponibilità specifica, consentono di prenotare capacità. Tuttavia, a parte la prenotazione della disponibilità e lo sconto sul prezzo, non vi è alcuna differenza di funzionalità tra le istanze riservate e le istanze on demand.

Le istanze Spot consentono di fare un'offerta sulla capacità di elaborazione libera di Amazon EC2. Le istanze Spot sono spesso disponibili a un prezzo inferiore rispetto a quello on demand e ciò consente di ridurre in modo significativo i costi dell'esecuzione delle applicazioni basate su cloud.

Durante la progettazione per il cloud, alcuni casi d'uso risultano più adatti per l'uso di istanze Spot rispetto ad altri. Ad esempio, poiché le istanze Spot possono essere ritirate in qualsiasi momento qualora il prezzo di offerta sia superiore all'offerta dell'utente, è necessario utilizzare le istanze Spot soltanto per stack di applicazioni relativamente stateless e dimensionabili orizzontalmente. Per le applicazioni stateful o per carichi di elaborazione particolarmente impegnativi, l'uso di istanze riservate o di istanze on demand potrebbe essere un'opzione più valida. Per le applicazioni mission critical, in cui le limitazioni di capacità non possono essere tollerate, le istanze riservate risultano la scelta ottimale.

Consultare Reserved Instances[17] e Spot Instances[18] per ulteriori dettagli.

Processo di gestione degli eventi

La pianificazione di un evento di infrastruttura è un'attività di gruppo che coinvolge gli sviluppatori dell'applicazione, gli amministratori e gli stakeholder aziendali. Alcune settimane prima di un evento di infrastruttura, è consigliabile tenere delle riunioni con cadenza ricorrente coinvolgendo lo staff tecnico che possiede e gestisce i componenti infrastrutturali chiave del servizio Web.

Pianificazione dell'evento di infrastruttura

La pianificazione di un evento di infrastruttura dovrebbe iniziare alcune settimane prima della data dell'evento. Una sequenza temporale tipica nel ciclo di vita di un evento pianificato è illustrata nella Figura 2.

Figura 2. Sequenza temporale tipica di eventi di infrastruttura

Pianificazione e preparazione

Pianificazione

Consigliamo di pianificare le seguenti attività nelle settimane precedenti a un evento di infrastruttura:

Settimana 1:

- Nomina di un team per guidare la pianificazione e la progettazione dell'evento di infrastruttura.
- Organizzazione di riunioni con gli stakeholder aziendali per comprendere i parametri dell'evento (dimensionamento, durata, orario, estensione geografica, carichi di lavoro coinvolti) e i criteri di determinazione del successo.
- Coinvolgimento di tutti i partner e fornitori a monte e a valle.

Settimane 2-3:

- Verifica dell'architettura e applicazione delle modifiche necessarie.
- Esecuzione di una revisione operativa; applicazione delle modifiche necessarie.
- Osservanza delle best practice descritte in questo documento e nei riferimenti a piè di pagina.

- Identificazione dei rischi e sviluppo di piani di attenuazione.
- Sviluppo di un runbook dell'evento pianificato.

Settimana 4:

- Esame di tutti i fornitori di servizi cloud che necessitano di dimensionamento in base al carico stimato.
- Controllo dei limiti del servizio e loro aumento in base alle esigenze.
- Configurazione di un pannello di controllo per il monitoraggio e degli avvisi al raggiungimento di soglie definite.

Revisione dell'architettura

Un componente essenziale per la preparazione di un evento di infrastruttura è una revisione dell'architettura dello stack di applicazioni che registrerà un aumento di traffico. Lo scopo della revisione è quello di verificare e identificare le potenziali aree di rischio per la scalabilità o l'affidabilità dell'applicazione, nonché identificare le opportunità per l'ottimizzazione anticipata dell'evento.

AWS offre ai clienti con contratto di supporto Enterprise una strategia per rivedere gli stack di applicazioni basata su cinque pilastri di progettazione: sicurezza, affidabilità, efficienza delle prestazioni, ottimizzazione dei costi ed eccellenza operativa.

Tabella 1. Pilastri di applicazioni ben realizzate

Nome pilastro	Definizione pilastro	Aree di interesse rilevanti
Sicurezza	La capacità di proteggere informazioni, sistemi e asset fornendo allo stesso tempo valore di business tramite valutazioni di rischio e strategie di mitigazione.	Gestione dell'identità, crittografia, monitoraggio, logging, gestione delle chiavi, istanze dedicate, conformità, governance
Affidabilità	La capacità di un sistema di recuperare da un errore di infrastruttura o di servizio, di acquisire dinamicamente le risorse di elaborazione per soddisfare la domanda e di mitigare interruzioni di servizio dovute a configurazioni errate o problemi di rete momentanei.	Limiti di servizio, zone di disponibilità e regioni multiple, scalabilità, controllo dello stato/monitoraggio, backup/disaster recovery, reti, ripristino automatico

Nome pilastro	Definizione pilastro	Aree di interesse rilevanti
Efficienza delle prestazioni	La capacità di utilizzare le risorse di elaborazione in modo efficiente per soddisfare i requisiti di sistema e per mantenere l'efficienza anche con le variazioni della domanda e l'evoluzione della tecnologia.	Servizi AWS corretti, utilizzo delle risorse, architettura di storage, caching, requisiti di latenza
Ottimizzazione dei costi	La capacità di evitare o eliminare i costi superflui e le risorse non ottimali.	Istanze riservate/Spot, ottimizzazione dell'ambiente, selezione dei servizi, ottimizzazione dei volumi, gestione degli account, fatturazione consolidata, ritiro delle risorse
Eccellenza operativa	La capacità di eseguire e monitorare i sistemi per ottenere valore commerciale e il continuo miglioramento di processi e procedure di supporto.	Runbook, playbook, CI/CD, Game Days, infrastruttura come codice, RCA

Un elenco dettagliato per la revisione dei componenti dell'architettura, che può essere utilizzato per rivedere lo stack di applicazioni basato su AWS, è disponibile nell'Appendice di questo whitepaper.

Revisione operativa

Oltre a effettuare una revisione dell'architettura, che è maggiormente incentrata sulla progettazione dei componenti di un'applicazione, è necessario rivedere le prassi operative e di gestione del cloud per valutare se si sta amministrando in modo corretto il carico di lavoro su cloud. L'obiettivo della revisione è identificare le lacune e i problemi operativi e intraprendere azioni opportune prima dell'evento per ridurne al minimo la portata.

AWS offre un servizio di revisione delle operazioni su cloud ai clienti con contratto di supporto Enterprise, che può essere uno strumento efficace per preparare un evento di infrastruttura. La revisione è incentrata sulla valutazione delle seguenti aree:

- Preparazione: è necessario disporre della giusta combinazione di struttura organizzativa, processi e tecnologie. È necessario definire ruoli chiari e responsabilità per il personale che gestisce lo stack di applicazioni. I processi devono essere definiti in anticipo per allinearli all'evento. Le procedure devono essere automatizzate laddove possibile.
- Monitoraggio: un monitoraggio efficace misura le prestazioni di un'applicazione. Il monitoraggio è cruciale per rilevare anomalie prima

che si trasformino in problemi e offre la possibilità di ridurre al minimo l'impatto di eventi avversi.

- Operazioni: le attività operative devono essere effettuate in modo tempestivo e affidabile sfruttando l'automazione ovunque possibile e affrontando allo stesso tempo eventi operativi inattesi che richiedono escalation.
- Ottimizzazione: un'analisi a posteriori deve essere effettuata utilizzando i parametri raccolti, i trend operativi e le lezioni apprese, in modo da acquisire e identificare le opportunità di miglioramento per eventi futuri. L'unione di ottimizzazione e preparazione crea un ciclo di feedback per affrontare problemi operativi ed evitare che si ripetano.

Limiti di servizio AWS

Durante la pianificazione di un evento di infrastruttura, è cruciale evitare di superare i limiti imposti dal provider di servizi cloud al dimensionamento di un'applicazione o di un carico di lavoro.

Normalmente, i provider di servizi cloud hanno dei limiti sulle varie risorse che è possibile utilizzare. Questi sono di solito imposti sulla base del tipo di account e della regione. Le risorse interessate comprendono istanze, volumi, stream, invocazioni serverless, snapshot, numero di VPC, regole di sicurezza e così via. Tali limiti sono ideati come misura di sicurezza contro codici runaway o tentativi di uso illecito delle risorse da parte di utenti malintenzionati e come sistema di controllo per ridurre al minimo i rischi di costi imprevisti.

Alcune limitazioni vengono automaticamente ridotte nel corso del tempo, man mano che il cliente espande il suo utilizzo del cloud. Tuttavia, per la maggior parte dei servizi occorre richiedere la riduzione delle limitazioni aprendo un caso di supporto. Altri servizi, invece, prevedono dei limiti che non possono essere modificati.

AWS mette a disposizione dei clienti con contratto di supporto Enterprise e Business il servizio Trusted Advisor, un pannello di controllo dei limiti che permette di gestire in modo proattivo tutti i limiti di servizio.

Per ulteriori informazioni sui limiti dei vari servizi di AWS e su come controllarli, consultare AWS Service Limits[19] e Trusted Advisor.[20]

Nozioni sui pattern

Valori standard

È consigliabile documentare i valori di "stato normale" per i parametri chiave prima di cominciare un evento di infrastruttura. Ciò aiuta a determinare quando un'applicazione o un servizio sono tornati a livelli normali al termine dell'evento. Ad esempio, se si determina che la frequenza normale delle transazioni attraverso un sistema di bilanciamento del carico è di 2.500 richieste al secondo, sarà più facile stabilire quando avviare le procedure di ridimensionamento una volta concluso l'evento.

Flussi dei dati e dipendenze

Comprendere il modo in cui si sviluppa il flusso di dati attraverso le varie componenti di un'applicazione aiuta a identificare i potenziali colli di bottiglia e le dipendenze. I livelli o i componenti di un'applicazione che utilizzano i dati in un flusso sono dimensionati e configurati correttamente per auto-dimensionarsi quando i livelli o i componenti di uno stack di applicazioni che generano dati aumentano di dimensione? In caso di malfunzionamento di un componente, i dati possono essere accodati fino al suo ripristino? I fornitori o gli utilizzatori dei dati a monte o a valle sono scalabili in relazione all'evento?

Proporzionalità

Un altro aspetto da tenere in considerazione nella preparazione di un evento di infrastruttura è la proporzionalità di dimensionamento richiesta dai vari componenti di uno stack di applicazioni. Questa proporzionalità non è sempre di tipo uno a uno. Ad esempio, un aumento di dieci volte del tasso di operazioni al secondo in un sistema di bilanciamento del carico potrebbe richiedere un incremento di venti volte della capacità di storage, del numero di shard di streaming o del numero di operazioni di lettura e scrittura del database, a causa delle attività di elaborazione in corso nell'applicazione front-end.

Piano di comunicazione

Prima dell'evento, è necessario sviluppare un piano di comunicazione. Compilare una lista degli stakeholder interni e dei gruppi di supporto e identificare chi contattare durante le varie fasi dell'evento nei vari scenari, quali l'inizio, lo svolgimento e la fine dell'evento, l'analisi a posteriori, contatti di emergenza, contatti durante situazioni che richiedono la risoluzione di problemi, ecc.

Le persone e i gruppi da contattare potranno includere:

- Stakeholder
- Responsabili delle operazioni
- Sviluppatori
- Team di supporto
- Team del provider di servizi cloud
- Team del centro operativo di rete (NOC)

Contemporaneamente a una lista dei contatti interni, è opportuno compilare una lista degli stakeholder esterni coinvolti nella distribuzione in tempo reale dell'applicazione. Questi stakeholder includono i partner e i fornitori che supportano i componenti chiave dello stack, i venditori a monte e a valle che forniscono i servizi esterni, i feed di dati, i servizi di autenticazione e così via.

Questa lista di contatti esterni deve inoltre includere:

- I fornitori di hosting dell'infrastruttura
- I provider delle telecomunicazioni
- I partner di streaming in tempo reale dei dati
- I contatti di PR e marketing
- I partner pubblicitari
- I consulenti tecnici coinvolti nella progettazione del servizio

A ciascun provider, richiedere le seguenti informazioni:

- Punti di contatto in tempo reale durante lo svolgimento dell'evento
- Contatto di supporto critico e processo di escalation
- Nome, numero di telefono e indirizzo e-mail
- Conferma che i contatti tecnici saranno disponibili in tempo reale

Agli account dei clienti AWS che hanno sottoscritto il piano di supporto Enterprise viene inoltre assegnato un Technical Account Manager (TAM), il quale può coordinare e verificare che lo staff di supporto AWS dedicato sia a conoscenza dell'evento e preparato a fornire assistenza. I TAM sono anche

disponibili durante l'evento, presenti nella sala operativa e, se necessario, possono occuparsi dell'escalation dei problemi.

Preparazione del centro operativo di rete

Prima dell'evento, è necessario istruire il team delle operazioni e/o di sviluppatori affinché creino un pannello di controllo con parametri in tempo reale per il monitoraggio di ogni componente critico del servizio Web in produzione quando l'evento è in corso. Idealmente, il pannello di controllo dovrebbe presentare automaticamente parametri aggiornati ad intervalli di un minuto o qualsiasi altro intervallo di tempo idoneo e rilevante durante l'evento.

È consigliabile monitorare i seguenti componenti:

- Utilizzo delle risorse di ogni server (CPU, memoria e dischi)
- Tempi di risposta del servizio Web
- Parametri di traffico Web (utenti, visualizzazioni di pagine, sessioni)
- Traffico Web per ciascuna regione dei visitatori (segmenti dei clienti globali)
- Utilizzo del server di database
- Funnel di conversione del flusso di marketing, ad esempio tassi di conversione e percentuali di fallout
- Log di errore delle applicazioni
- Indicatori preventivi di errore

Amazon CloudWatch fornisce uno strumento per raccogliere la maggior parte di questi parametri provenienti dalle risorse AWS in un'unica console attraverso pannelli di controllo personalizzabili. Inoltre, CloudWatch consente di importare parametri personalizzati ogni qual volta AWS non fornisca tali parametri automaticamente. Consultare la sezione Monitoraggio di questo whitepaper per ulteriori dettagli sulle funzionalità degli strumenti di monitoraggio AWS.

Preparazione di un runbook

È consigliabile redigere un runbook in preparazione dell'evento di infrastruttura. Un *runbook* è un manuale operativo contenente una serie di procedure e operazioni che gli addetti dovranno svolgere durante l'evento. I

runbook dell'evento possono essere un'estensione dei runbook esistenti utilizzati per le operazioni di routine e la gestione delle eccezioni. Di solito, un runbook contiene le procedure per avviare, arrestare, supervisionare ed eseguire il debugging di un sistema. Descrive inoltre le procedure per la gestione di eventi inattesi e di contingenze.

Un runbook deve includere le seguenti sezioni:

- **Dettagli dell'evento**: breve descrizione dell'evento, criteri di successo, copertura mediatica, date dell'evento e contatti dettagliati dei principali stakeholder del cliente e di AWS.
- **Elenco dei servizi AWS**: enumera tutti i servizi AWS che saranno utilizzati durante l'evento. Indica inoltre il carico previsto su questi servizi, le regioni coinvolte e gli ID degli account.
- **Revisione dell'architettura e dell'applicazione**: documenta i risultati dei test di carico, eventuali punti di stress nell'architettura dell'infrastruttura e dell'applicazione, le misure di resilienza per il carico di lavoro, i singoli punti di errore e i potenziali colli di bottiglia.
- **Revisione operativa**: mostra la configurazione di monitoraggio, i criteri di stato, i meccanismi di notifica e le procedure di ripristino del servizio.
- **Lista di controllo della preparazione**: include considerazioni quali: controlli sui limiti di servizio, pre-caricamento dei componenti dello stack di applicazioni come il sistema di bilanciamento del carico, pre-provisioning delle risorse come gli shard di streaming, partizioni DynamoDB, partizioni S3 e così via. Per ulteriori informazioni, consultare la Lista di controllo dettagliata per la revisione dell'architettura nell'Appendice di questo whitepaper.

Monitoraggio

Piano di monitoraggio

Il monitoraggio del database, dell'applicazione e del sistema operativo è un aspetto cruciale per garantire il successo dell'evento. È consigliabile configurare sistemi di monitoraggio completi in modo da poter rilevare efficacemente e rispondere immediatamente a incidenti di elevata gravità nel corso dell'evento di infrastruttura. A livello generale, una strategia di monitoraggio efficace deve garantire che gli strumenti di monitoraggio siano dotati delle caratteristiche

adeguate all'applicazione in base alla sua criticità per l'azienda. Una strategia efficace di gestione degli incidenti incorporerà sia i dati di monitoraggio di AWS che quelli del cliente insieme agli strumenti e ai processi di gestione degli eventi e degli incidenti. L'implementazione di un piano di monitoraggio che raccolga tutti i dati provenienti da tutti i segmenti della soluzione AWS sarà di enorme aiuto nel debugging di eventuali errori complessi.

Il piano di monitoraggio dovrebbe rispondere alle seguenti domande:

- Quali strumenti di monitoraggio e pannelli di controllo devono essere configurati per l'evento?
- Quali sono gli obiettivi del monitoraggio e le soglie consentite? Quali eventi attiveranno azioni?
- Quali risorse e quali parametri di queste risorse saranno monitorati e con quale frequenza devono essere raccolte le informazioni?
- Chi eseguirà le attività di monitoraggio? Quali avvisi di monitoraggio sono attivi? Chi riceverà gli avvisi?
- Quali piani di intervento sono stati allestiti per gestire errori comuni e previsti? Quali misure sono previste per gli eventi inattesi?
- Qual è il processo di escalation in caso di malfunzionamenti?

Nell'ambito di questa strategia, possono essere utilizzati i seguenti strumenti di monitoraggio di AWS:

- **Amazon CloudWatch**: una soluzione pronta che fornisce una dashboard di controllo per i parametri, il monitoraggio, gli avvisi e il provisioning automatizzato di AWS.
- **Parametri personalizzati di Amazon CloudWatch**: utilizzati per raccogliere i parametri dai sistemi operativi, dalle applicazioni e dall'azienda. Le API di Amazon CloudWatch consentono di raccogliere praticamente qualsiasi tipo di parametro personalizzato.
- **Stato dell'istanza EC2 Amazon**: consente di visualizzare controlli di stato e per pianificare eventi, come il riavvio automatico di un'istanza, per le istanze in base al loro stato.
- **Amazon SNS**: permette di configurare, gestire e inviare notifiche basate sugli eventi.

- **AWS X-Ray**: aiuta nelle operazioni di debugging e di analisi di applicazioni distribuite e di architetture di microservizi attraverso l'analisi dei flussi di dati tra i vari componenti del sistema.
- **Servizio Amazon Elasticsearch**: utilizzato per la raccolta centralizzata dei log e la loro analisi in tempo reale. Consente una rilevamento euristico e rapido dei problemi.
- **Strumenti di terze parti**: utilizzati per le analisi in tempo reale e per il monitoraggio e la visibilità dell'intero stack.
- **Strumenti di monitoraggio standard del sistema operativo**: consentono il monitoraggio a livello di sistema operativo.

Per ulteriori informazioni sugli strumenti di monitoraggio AWS, consultare Automated and Manual Monitoring.[21] Consultare anche Using Amazon CloudWatch Dashboards[22] e Publishing Custom Metrics.[23]

Notifiche

Un elemento operativo cruciale nella progettazione di un evento di infrastruttura è la configurazione di allarmi e notifiche da integrare nelle soluzioni di monitoraggio. Questi allarmi e notifiche possono essere utilizzati con servizi quali AWS Lambda per attivare azioni in base agli allarmi. L'automazione delle risposte a eventi operativi è un elemento chiave per attuare procedure di mitigazione, rollback e ripristino con la massima reattività.

Inoltre, è necessario implementare strumenti per monitorare centralmente i carichi di lavoro e creare avvisi e notifiche appropriati basati sui log disponibili e sui parametri legati agli indicatori operativi più importanti, tra cui allarmi e notifiche relative ad anomalie fuori intervallo, nonché a malfunzionamenti nel servizio o nei componenti. Idealmente, quando vengono superate determinate soglie di prestazioni minime o si verificano errori, l'architettura del sistema è stata progettata per eseguire un ripristino automatico o per ridimensionare le risorse in risposta a tali notifiche e avvisi.

Come sottolineato in precedenza, AWS offre servizi (Amazon SQS e Amazon SNS) per garantire avvisi e notifiche appropriati in risposta a eventi operativi imprevisti, nonché per abilitare risposte automatiche.

Prontezza operativa (giorno dell'evento)

Esecuzione del piano

Il giorno dell'evento, il team principale coinvolto nell'evento di infrastruttura dovrebbe essere connesso in teleconferenza per monitorare i pannelli di controllo in tempo reale. I runbook devono essere stati completamente redatti ed essere disponibili. Accertarsi che il piano di comunicazione sia ben definito e noto a tutti i membri dello staff di supporto e agli stakeholder, e che sia pronto un piano di emergenza.

Sala operativa

Durante l'evento, è necessario avere un collegamento aperto in teleconferenza con i seguenti partecipanti:

- Il responsabile principale dell'applicazione e i team operativi
- I responsabili del team operativo
- Le risorse tecniche dai partner esterni direttamente coinvolte con la fornitura tecnica
- Gli stakeholder aziendali

Durante gran parte dell'evento, la conversazione attraverso questo canale di collegamento deve essere minima. Se si verifica un evento operativo avverso, le persone chiave che possono far fronte a tale circostanza saranno già pronte ad agire e a consultarsi attraverso questo canale di collegamento.

Report direttivo

Durante l'evento, inviare un'e-mail ogni ora agli stakeholder del direttivo. Questo aggiornamento deve includere le seguenti informazioni:

- Riepilogo dello stato: verde (secondo i piani), giallo (riscontrati problemi), rosso (problemi di gravità elevata)
- Aggiornamento sui parametri chiave
- Problemi rilevati, stato del piano di risoluzione, tempo previsto per la risoluzione
- Numero di telefono del canale di collegamento della sala operativa (nel caso qualcuno voglia partecipare)

Alla conclusione dell'evento, inviare un riepilogo finale via e-mail in un formato analogo.

Piano di emergenza

Ogni fase del processo di preparazione all'evento deve avere un piano corrispondente di rollback già verificato in un ambiente di test.

Durante la stesura di un piano di rollback, considerare i seguenti aspetti:

- Quali sono gli scenari peggiori che possono verificarsi durante l'evento?
- Quale tipo di eventi potrebbe provocare un impatto negativo sulle relazioni con il pubblico?
- Quali servizi e componenti di terze parti potrebbero subire dei malfunzionamenti durante l'evento?
- Quali parametri che indicherebbero il verificarsi di uno scenario negativo dovrebbero essere monitorati?
- Qual è il piano di rollback per ogni scenario possibile?
- Quanto tempo richiederà ciascun processo di rollback? Quali sono gli obiettivi di punto di ripristino (RPO) e di tempo di ripristino (RTO) accettabili? Consultare Using AWS for Disaster Recovery[24] per approfondimenti su queste nozioni.

Prendere in considerazione i seguenti tipi di rollback:

- **Distribuzione blu/verde**: se si distribuisce una nuova applicazione o un nuovo ambiente di produzione, mantenere la build di produzione precedente in linea e pronta per essere ripristinata rapidamente.
- **Warm pilot**: avviare in una seconda regione un ambiente minimo che possa essere dimensionato velocemente all'occorrenza. Se la regione primaria non è disponibile, ricalibrare velocemente l'ambiente di produzione nella regione di backup e reindirizzare il traffico verso la seconda regione.
- **Pagine di errore modalità di manutenzione**: controllare le funzionalità e i trigger della pagina di errore per ciascun layer del servizio Web. Prepararsi a inserire in queste pagine di errore un messaggio più specifico, se necessario.

Testare e documentare ciascun piano di rollback per ogni possibile scenario di malfunzionamento.

Attività post-evento

Analisi a posteriori

Un'analisi a posteriori viene troppo spesso trascurata perché i clienti sono in genere ansiosi di tornare all'operatività normale. Tuttavia, consigliamo di esigere un'analisi a posteriori come parte essenziale della gestione di qualsiasi evento di infrastruttura. L'analisi a posteriori permette di collaborare con ogni team coinvolto e di identificare aree che potrebbero richiedere ulteriore ottimizzazione, come procedure operative, dettagli di implementazione, procedure di failover e di ripristino e così via. Questo requisito è particolarmente importante nel caso in cui lo stack di applicazioni smetta di funzionare durante un evento. Un'analisi a posteriori dell'evento aiuterà anche a produrre la documentazione necessaria per l'eventuale RCA (analisi delle cause primarie).

Processo di ridimensionamento

Immediatamente dopo la conclusione dell'evento di infrastruttura, deve iniziare il processo di ridimensionamento. Durante tale periodo, è consigliabile continuare a monitorare le applicazioni e i servizi coinvolti per garantire che il traffico venga riportato ai livelli di produzione normali. Utilizzare i pannelli di controllo dello stato creati durante la fase di preparazione per verificare che il traffico e la frequenza delle transazioni si siano normalizzati. Per alcuni eventi, i periodi di ridimensionamento possono essere lineari e semplici, mentre altri potrebbero subire riduzioni di volume irregolari o più graduali. Alcuni pattern di traffico potrebbero persistere. Ad esempio, la ripresa da un picco di traffico richiede di solito procedure di ridimensionamento lineari, mentre eventi come la distribuzione di un'applicazione o l'espansione in una nuova regione geografica potrebbero avere degli effetti durevoli, che obbligano a monitorare attentamente i nuovi pattern di traffico e a introdurre strumenti di monitoraggio aggiuntivi nello stack di applicazioni permanente.

A un certo punto dopo il completamento dell'evento è necessario determinare quando è possibile concludere in modo sicuro le operazioni di gestione dell'evento. Fare riferimento ai valori "normali" dei parametri chiave precedentemente documentati per determinare quando è possibile dichiarare un evento completato o terminato. È consigliabile suddividere le attività di

ridimensionamento in due direzioni, con possibili sequenze temporali diverse. Concentrare la prima direzione sulla gestione operativa dell'evento, come l'invio di comunicazioni agli stakeholder interni ed esterni e ai partner e la reimpostazione dei limiti di servizio. Concentrare la seconda direzione sugli aspetti tecnici del ridimensionamento, come procedure di ridimensionamento, convalida dello stato dell'ambiente e criteri per determinare se le modifiche all'architettura possano essere revocate o mantenute.

La sequenza temporale associata a ciascuna di queste direzioni può variare a seconda della natura dell'evento, dei parametri chiave e del grado di soddisfazione del cliente. Nella tabella riportata di seguito abbiamo delineato alcune attività comuni associate a ciascuna direzione come guida di riferimento per determinare la gestione più appropriata del periodo finale di un evento.

Tabella 2. Attività operative di ridimensionamento

Attività	Descrizione
Comunicazioni	Notifica agli stakeholder interni ed esterni che l'evento è terminato. Le comunicazioni durante il periodo finale devono essere coerenti con la definizione di completamento dell'evento. Utilizzare i parametri di "stato normale" per determinare quando è opportuno cessare le comunicazioni. In alternativa, è possibile chiudere le comunicazioni in più tempi. Ad esempio, è possibile chiudere il canale di collegamento della sala operativa ma lasciare intatte le procedure di escalation in caso di problemi post-evento.
Limiti di servizio/ contenimento dei costi	Anche se può essere allettante mantenere un limite di servizio elevato dopo un evento, tenere presente che i limiti di servizio esistono anche per motivi di sicurezza. I limiti di servizio proteggono l'utente prevenendo costi dovuti a un uso eccessivo del servizio, sia questo dovuto alla compromissione dell'account o a una funzione automatica configurata in modo errato.
Reporting e analisi	È opportuno effettuare una raccolta dei dati e un confronto dei parametri dell'evento, accompagnati da un resoconto analitico di pattern, trend, aree problematiche, procedure riuscite, procedure ad-hoc, sequenza temporale degli eventi e soddisfacimento o meno dei criteri di successo. Queste informazioni devono poi essere distribuite a tutte le parti identificate nel piano di comunicazione. Inoltre va condotta un'analisi dei costi dettagliata che evidenzi le spese operative sostenute per il supporto dell'evento.
Attività di ottimizzazione	Le grandi organizzazioni si evolvono nel tempo continuando a migliorare le proprie operazioni. L'ottimizzazione operativa richiede la raccolta costante di parametri, tendenze operative e lezioni apprese da eventi passati per rivelare opportunità di miglioramento. L'ottimizzazione va di pari passo con la preparazione per formare un ciclo di feedback, in modo da affrontare i problemi operativi ed evitare che si ripetano.

Tabella 3. Attività tecniche di ridimensionamento

Attività	Descrizione
Limiti di servizio/ contenimento dei costi	Anche se può essere allettante mantenere un limite di servizio elevato dopo un evento, tenere presente che i limiti di servizio esistono anche per motivi di sicurezza. I limiti di servizio proteggono l'utente prevenendo costi dovuti a un uso eccessivo del servizio, sia questo dovuto alla compromissione dell'account o a una funzione automatica configurata in modo errato.
Procedure di ridimensionamento	Ripristinare le risorse potenziate durante la fase di preparazione. Questi elementi sono specifici di ciascuna architettura, ma i seguenti esempi sono comuni: Dimensioni delle istanze EC2/RDS Configurazione di Auto Scaling Capacità prenotata IOPS configurati
Convalida dell'integrità dell'ambiente	Confrontare i parametri standard ed esaminare lo stato della produzione per verificare che dopo la conclusione dell'evento e delle procedure di ridimensionamento i sistemi coinvolti indicano un funzionamento normale.
Mantenimento delle modifiche all'architettura	Potrebbe essere opportuno mantenere alcune modifiche apportate durante la preparazione dell'evento, a seconda della natura dell'evento stesso e dei valori dei parametri operativi emersi. Ad esempio, l'espansione in una nuova area geografica potrebbe richiedere un aumento permanente delle risorse in quella regione; oppure l'aumento dei limiti di determinati servizi o parametri di configurazione, come il numero di partizioni in un database o il numero di shard in un flusso di PIOPS in un volume, potrebbe essere un'ottimizzazione delle prestazioni che è opportuno mantenere.

Ottimizzazione

Probabilmente, il componente più importante nella gestione dell'evento di infrastruttura è l'analisi post-evento e l'identificazione delle sfide operative e architetturali osservate insieme alle possibili opportunità di miglioramento. Gli eventi di infrastruttura sono raramente eventi isolati. Essi possono essere stagionali o coincidere con nuove versioni di un'applicazione, oppure possono far parte della crescita dell'azienda nel corso della sua espansione in nuovi mercati e aree geografiche. Pertanto, ogni evento di infrastruttura rappresenta un'opportunità per osservare, migliorare e preparare più efficacemente l'evento successivo.

Conclusioni

AWS fornisce moduli base sotto forma di prodotti elastici e programmabili e servizi che l'azienda può assemblare per supportare praticamente qualsiasi

livello di carico di lavoro. Grazie alle linee guida e best practice di AWS per la gestione degli eventi di infrastruttura, unite al nostro set completo di servizi altamente disponibili, le aziende possono preparare eventi di grande portata con la certezza di poter gestire le esigenze di dimensionamento senza intoppi e in maniera dinamica, garantendo una risposta rapida e una portata globale.

Collaboratori

Alla stesura di questo documento hanno collaborato le persone e le organizzazioni indicate di seguito:

- Presley Acuna, AWS Enterprise Support Manager
- Kurt Gray, AWS Global Solutions Architect
- Michael Bozek, AWS Sr. Technical Account Manager
- Rovan Omar, AWS Technical Account Manager
- Will Badr, AWS Technical Account Manager
- Eric Blankenship, AWS Sr. Technical Account Manager
- Greg Bur, AWS Technical Account Manager
- Bill Hesse, AWS Sr. Technical Account Manager
- Hasan Khan, AWS Sr. Technical Account Manager
- Varun Bakshi, AWS Sr. Technical Account Manager

Approfondimenti

Per approfondimenti sulle best practice operative e architetturali, consultare Operational Checklist for AWS.[25] Si consiglia di leggere AWS Well Architected Framework[26] per un approccio strutturato alla valutazione degli stack di distribuzione di applicazioni basati su cloud. AWS offre Infrastructure Event Management (IEM) come opzione di supporto avanzato per i clienti che desiderano un coinvolgimento più diretto dei Technical Account Manager AWS e dei tecnici di supporto nelle operazioni di progettazione, pianificazione e nel giorno dell'evento. Per ulteriori informazioni sull'offerta del supporto avanzato IEM di AWS, consultare Infrastructure Event Management.[27]

Appendice

Lista di controllo dettagliata per la revisione dell'architettura

Sì-No-N/A	Sicurezza
☐-☐-☐	Modifichiamo ogni 3 mesi le nostre chiavi di accesso e le password utente di AWS Identity and Access Management (IAM), nonché le credenziali per le risorse appartenenti alla nostra applicazione, come richiesto dalle best practice di sicurezza di AWS. Applichiamo questa policy di password in ogni account e utilizziamo dispositivi hardware o virtuali per l'autenticazione multifattoriale (MFA).
☐-☐-☐	Disponiamo di processi di sicurezza interna e meccanismi di controllo degli accessi alle API di AWS tramite IAM, utilizzando tipologie di accesso univoche, basate sul ruolo e a privilegio minimo.
☐-☐-☐	Abbiamo rimosso tutte le informazioni sensibili o riservate, incluse le coppie di chiavi pubbliche e private di accesso alle istanze e abbiamo revisionato tutti i file delle chiavi SSH autorizzate per qualsiasi Amazon Machine Image (AMI) personalizzata.
☐-☐-☐	Utilizziamo ruoli IAM per le istanze EC2 quando possibile, anziché incorporare le credenziali all'interno delle AMI.
☐-☐-☐	Abbiamo separato i privilegi amministrativi IAM da quelli degli utenti normali mediante la creazione di un ruolo amministrativo IAM, limitando dagli altri ruoli funzionali le operazioni IAM.
☐-☐-☐	Applichiamo le patch di sicurezza più recenti sulle nostre istanze EC2 sia per Windows che per Linux. Utilizziamo controlli di accesso ai sistemi operativi, tra cui regole Amazon EC2 Security Group, liste di controllo dell'accesso alla rete VPC, protezione avanzata dei sistemi operativi, firewall basati su host, rilevamento e prevenzione delle intrusioni, configurazione dei software di monitoraggio e inventario host.
☐-☐-☐	Garantiamo che la connettività di rete da e verso gli ambienti AWS e aziendali dell'organizzazione utilizzino uno scambio di protocolli di crittografia.
☐-☐-☐	Adottiamo una soluzione di gestione centralizzata dei log e degli audit per identificare e analizzare modelli di accesso insoliti o qualsiasi attacco dannoso all'ambiente.
☐-☐-☐	Abbiamo implementato una gestione degli eventi e degli incidenti di sicurezza, correlazione, nonché processi di reporting.
☐-☐-☐	Ci assicuriamo che non vi siano accessi illimitati alle risorse AWS in ognuno dei nostri gruppi di sicurezza.
☐-☐-☐	Utilizziamo protocolli di sicurezza (HTTPS o SSL), policy di sicurezza e protocolli di crittografia aggiornati per la connessione al front-end (dal client al sistema di bilanciamento del carico). Le richieste vengono crittografate tra i client e il sistema di bilanciamento del carico, rendendo il processo più sicuro.
☐-☐-☐	Configuriamo il nostro set di record di risorse MX di Amazon Route 53 in modo che mantenga un set di record TXT di risorse, contenente un valore di Sender Policy Framework (SPF) corrispondente per specificare i server autorizzati a inviare e-mail dal nostro dominio.

Sì-No-N/A	Affidabilità
☐-☐-☐	Distribuiamo la nostra applicazione attraverso una flotta di istanze EC2 che vengono implementate in un gruppo Auto Scaling per assicurare un dimensionamento orizzontale automatico basato su piani di dimensionamento predefiniti. Ulteriori informazioni.
☐-☐-☐	Utilizziamo un controllo dello stato tramite Elastic Load Balancing nella configurazione del gruppo Auto Scaling per assicurare che quest'ultimo agisca sull'integrità delle istanze EC2 sottostanti (applicabile solo se si utilizzano sistemi di bilanciamento del carico nei gruppi Auto Scaling).
☐-☐-☐	Distribuiamo i componenti critici delle nostre applicazioni su zone di disponibilità multiple, replicando opportunamente i dati tra le varie zone. Testiamo come eventuali malfunzionamenti all'interno di questi componenti possano influire sulla disponibilità dell'applicazione utilizzando Elastic Load Balancing, Amazon Route 53 o qualsiasi strumento appropriato di terze parti.
☐-☐-☐	Nel livello di database, distribuiamo le nostre istanze Amazon RDS su zone di disponibilità multiple per migliorare la disponibilità del database, replicando in maniera sincrona i dati in un'istanza di standby situata in una zona di disponibilità diversa.
☐-☐-☐	Abbiamo definito processi per failover automatici o manuali, nel caso si verifichi un'interruzione o un degrado delle prestazioni.
☐-☐-☐	Utilizziamo i record CNAME per mappare il nostro indirizzo DNS ai nostri servizi. NON utilizziamo record di tipo A.
☐-☐-☐	Abbiamo configurato un valore di durata (TTL) più basso per il nostro set di record di Amazon Route 53. Questo evita ritardi quando i risolutori DNS richiedono dei record DNS aggiornati durante il reindirizzamento del traffico (ciò può verificarsi ad esempio quando il failover DNS rileva e risponde a un errore di uno degli endpoint).
☐-☐-☐	Disponiamo di almeno due tunnel VPN, configurati per fornire ridondanza in caso di interruzioni o manutenzione pianificata dei dispositivi presso l'endpoint AWS.
☐-☐-☐	Utilizziamo AWS Direct Connect e disponiamo di due connessioni Direct Connect configurate in qualsiasi momento per fornire ridondanza nel caso un dispositivo non fosse disponibile. Le connessioni sono assegnate a ubicazioni Direct Connect diverse per fornire ridondanza nel caso una di queste non fosse disponibile. Abbiamo inoltre configurato la connettività verso i nostri gateway privati virtuali in modo da avere più interfacce virtuali configurate su più connessioni e ubicazioni Direct Connect.
☐-☐-☐	Utilizziamo istanze di Windows, assicurandoci che stiano utilizzando i driver PV più recenti. I driver PV aiutano a ottimizzare le prestazioni dei driver e a ridurre al minimo problemi di runtime e rischi di sicurezza. Ci siamo assicurati, inoltre, che nella nostra istanza di Windows sia in esecuzione la versione più recente dell'agente EC2Config.
☐-☐-☐	Creiamo degli snapshot dei nostri volumi Amazon Elastic Block Store (EBS) per garantire un ripristino point-in-time in caso di malfunzionamento.
☐-☐-☐	Utilizziamo volumi Amazon EBS distinti per il sistema operativo e per i dati dell'applicazione e dei database, in base alle esigenze.
☐-☐-☐	Abbiamo applicato le patch del kernel, del software e dei driver più recenti su tutte le istanze Linux.

Sì-No-N/A	Efficienza delle prestazioni
☐-☐-☐	Abbiamo effettuato dei test completi sui componenti della nostra applicazione in hosting su AWS, inclusi test delle prestazioni, prima di rilasciare l'applicazione. Abbiamo inoltre eseguito dei test di carico per assicurarci di aver utilizzato le istanze EC2 della dimensione adeguata, nonché il numero corretto di IOPS, la dimensione corretta dell'istanza database RDS, ecc.
☐-☐-☐	Generiamo un report di controllo dell'utilizzo rispetto ai nostri limiti di servizio per accertarci che l'attuale utilizzo su tutti i servizi AWS sia pari o inferiore all'80% dei limiti di servizio. Ulteriori informazioni
☐-☐-☐	Utilizziamo una rete di distribuzione dei contenuti (CDN) per effettuare il caching per la nostra applicazione (Amazon CloudFront) e come metodo per ottimizzare la diffusione dei contenuti e la loro distribuzione automatica sulla posizione periferica più vicina all'utente.
☐-☐-☐	Siamo consapevoli che alcuni header di richieste HTTP dinamiche che Amazon CloudFront riceve (agente utente, data, ecc.) possono influire sulle prestazioni, riducendo il rapporto di hit della cache e aumentando il carico sul server di origine. Ulteriori informazioni
☐-☐-☐	Ci assicuriamo che il throughput massimo di un'istanza EC2 sia superiore al throughput massimo totale dei volumi EBS collegati. Utilizziamo inoltre istanze ottimizzate per EBS con volumi PIOPS EBS per ottenere dai volumi le prestazioni previste.
☐-☐-☐	Ci assicuriamo che la progettazione della soluzione non presenti colli di bottiglia nell'infrastruttura o punti di stress nel database o nella struttura dell'applicazione.
☐-☐-☐	Implementiamo il monitoraggio delle risorse dell'applicazione e configuriamo degli allarmi basati su prestazioni anomale tramite Amazon CloudWatch o strumenti di partner di terze parti.
☐-☐-☐	La nostra progettazione evita l'utilizzo di un numero elevato di regole in qualsiasi gruppo di sicurezza collegato alle istanze della nostra applicazione. Un elevato numero di regole in un gruppo di sicurezza potrebbe degradare le prestazioni.

Sì-No-N/A	Ottimizzazione dei costi
☐-☐-☐	Siamo consapevoli che l'evento di infrastruttura può richiedere il provisioning di capacità extra, le quali devono essere rimosse una volta concluso l'evento, al fine di evitare spese superflue.
☐-☐-☐	Utilizziamo un dimensionamento ottimale per tutti i componenti della nostra infrastruttura, inclusi dimensione delle istanze EC2, dell'istanza di database RDS e dei volumi EBS, la dimensione e il numero di nodi dei cluster di caching e dei nodi del cluster Redshift.
☐-☐-☐	Utilizziamo le istanze Spot nei casi in cui risulta utile. Le istanze Spot sono ideali per carichi di lavoro che hanno tempi flessibili di inizio e fine. Tipici casi d'uso per le istanze Spot sono: elaborazione batch, generazione di report e carichi di lavoro HPC (High Performance Computing).

Sì-No-N/A	Ottimizzazione dei costi
☐-☐-☐	Abbiamo requisiti minimi prevedibili di capacità dell'applicazione e sfruttiamo i vantaggi offerti dalle istanze riservate. . Le istanze riservate consentono di prenotare capacità di calcolo con Amazon EC2 ottenendo un notevole sconto sulla tariffa oraria rispetto ai prezzi delle istanze on demand.

Notes

[1] https://aws.amazon.com/answers/account-management/aws-tagging-strategies/

[2] https://aws.amazon.com/blogs/aws/resource-groups-and-tagging/

[3] https://aws.amazon.com/sqs/

[4] http://docs.aws.amazon.com/general/latest/gr/rande.html

[5] https://aws.amazon.com/emr/

[6] https://aws.amazon.com/rds/

[7] https://aws.amazon.com/ecs/

[8] https://aws.amazon.com/sns/

[9] https://aws.amazon.com/blogs/compute/using-aws-lambda-with-auto-scaling-lifecycle-hooks/

[10] http://docs.aws.amazon.com/lambda/latest/dg/welcome.html

[11] https://aws.amazon.com/blogs/aws/new-auto-recovery-for-amazon-ec2/

[12] https://aws.amazon.com/answers/configuration-management/aws-infrastructure-configuration-management/

[13] https://d0.awsstatic.com/whitepapers/Big_Data_Analytics_Options_on_AWS%20.pdf

[14] http://docs.aws.amazon.com/Route53/latest/DeveloperGuide/routing-policy.html#routing-policy-latency

[15] https://aws.amazon.com/elasticache/

[16] https://aws.amazon.com/cloudfront/

[17] http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts-on-demand-reserved-instances.html

[18] http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-spot-instances.html

[19] https://docs.aws.amazon.com/general/latest/gr/aws_service_limits.html

[20] https://aws.amazon.com/about-aws/whats-new/2014/07/31/aws-trusted-advisor-security-and-service-limits-checks-now-free/

[21] http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/monitoring_automated_manual.html

[22] http://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/CloudWatch_Dashboards.html

[23] http://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/publishingMetrics.html

[24] https://aws.amazon.com/blogs/aws/new-whitepaper-use-aws-for-disaster-recovery/

[25] http://media.amazonwebservices.com/AWS_Operational_Checklists.pdf

[26] http://d0.awsstatic.com/whitepapers/architecture/AWS_Well-Architected_Framework.pdf

[27] https://aws.amazon.com/premiumsupport/iem/