Aumentare la resilienza e migliorare l'esperienza del cliente utilizzando l'ingegneria del caos su AWS

AWS Linee guida prescrittive

AWS Linee guida prescrittive: Aumentare la resilienza e migliorare l'esperienza del cliente utilizzando l'ingegneria del caos su AWS

Table of Contents

Aumentare la resilienza e migliorare l'esperienza del cliente utilizzando l'ingegneria del caos su AWS

Laurent Domb, responsabile tecnico, Federal Financials, Amazon Web Services

Aprile 2025 (storia del documento)

L'ingegneria del caos è la disciplina che consiste nella sperimentazione su un'applicazione per aumentare la fiducia nella capacità dell'organizzazione e dell'applicazione di resistere a condizioni di produzione turbolente. Si tratta di un approccio proattivo alla resilienza, con l'obiettivo di verificare se l'applicazione e l'organizzazione sono in grado di assorbire, adattarsi e infine riprendersi dai problemi del servizio introducendo guasti controllati tra persone, processi e tecnologie. L'intento è anche quello di identificare ed eliminare i punti deboli prima che possano causare interruzioni o altre interruzioni della produzione.

In Amazon, sappiamo che il fallimento è inevitabile nei sistemi distribuiti, al punto che funzionare nonostante la presenza di guasti è una modalità operativa normale. Poiché le interazioni tra i servizi sono destinate a fallire, è necessario comprendere come reagiscono i servizi durante le varie modalità di errore e creare servizi resilienti a vulnerabilità chiave come errori di dipendenza, tempeste di tentativi, zone di disponibilità compromesse e esaurimento delle risorse dell'host.

Prendiamo l'esempio di un Retry Storm. Un errore localizzato in un client può avere un impatto significativo su più servizi. Questo effetto viene comunemente chiamato effetto farfalla. Una tempesta di tentativi è una manifestazione dell'effetto farfalla, in cui una dipendenza non funzionante spinge i client, e i client di tali client, a riprovare l'operazione fallita, con conseguente crescita esponenziale del traffico. I servizi si sovraccaricano perché devono rispondere al traffico regolare oltre a ripetere il traffico, gestendo al contempo un peggioramento delle prestazioni.

L'ingegneria del caos è emersa come risposta alla crescente complessità dei sistemi distribuiti. È un approccio multidisciplinare che combina i principi della teoria del caos, del pensiero sistemico e dell'ingegneria per progettare e gestire sistemi complessi resistenti a eventi e comportamenti imprevisti. Fondamentalmente, l'ingegneria del caos si occupa della comprensione e della gestione del comportamento di sistemi complessi in condizioni di incertezza e imprevedibilità. Riconosce che gli approcci tradizionali all'ingegneria, che si basano sulla previsione e sul controllo dei risultati, sono spesso insufficienti per affrontare la natura complessa e dinamica dei sistemi distribuiti. Man mano che questi sistemi crescono, spesso superano l'ambito di comprensione di ogni singolo individuo.

L'ingegneria del caos fornisce concetti, tecniche e strumenti per iniettare intenzionalmente i guasti nei sistemi e scoprire i punti deboli prima che si manifestino in produzione. Questo approccio proattivo consente alle organizzazioni di acquisire la certezza che i propri sistemi funzioneranno in condizioni di stress. Sebbene l'ingegneria del caos sia ancora una pratica in evoluzione, rappresenta un cambiamento fondamentale verso la progettazione, la gestione e il funzionamento dei sistemi informatici moderni per renderli resilienti di fronte alla crescente complessità e interconnessione.

Le seguenti sezioni di questa guida illustrano i vantaggi dell'ingegneria del caos, spiegano come condurre esperimenti di ingegneria del caos e descrivono gli approcci che è possibile adottare per implementare l'ingegneria del caos su larga scala all'interno dell'organizzazione. Sono inclusi anche esempi di documenti di pianificazione degli esperimenti e dei risultati degli esperimenti che è possibile utilizzare come modelli per gli esperimenti di ingegneria del caos.

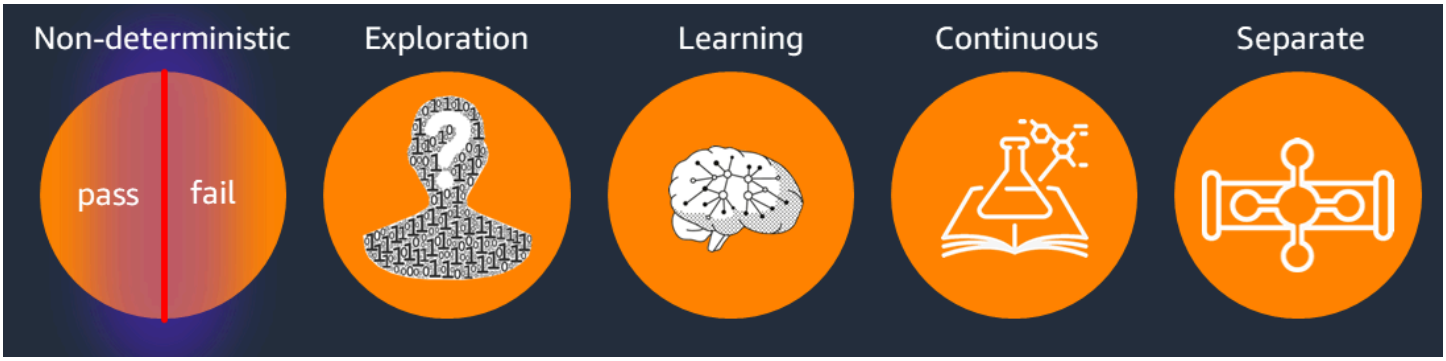
La sezione successiva esplora in che modo le caratteristiche dell'ingegneria del caos differiscono dai tradizionali test di resilienza come i test di unità, di fumo o di integrazione.

Panoramica

Confronto tra test di resilienza e ingegneria del caos

I test di resilienza sono deterministici. In altre parole, convalida le caratteristiche note dei meccanismi di resilienza, come interruttori automatici, nuovi tentativi, failover o fallback, che sono stati implementati nell'applicazione. Conferma come questi componenti dell'applicazione assorbano le interruzioni controllate con un impatto minimo o nullo sugli utenti. Pertanto, i test di resilienza si concentrano sulla convalida di modalità di errore note che vengono inserite nei componenti dell'applicazione con l'obiettivo di produrre risultati. pass/fail È consigliabile eseguire continuamente i test di resilienza come fase della pipeline per assicurarsi di non introdurre regressioni al proprio livello di resilienza. Nei test di resilienza, spesso non si eseguono test su componenti reali, ma si simulano un determinato componente. APIs Questo approccio consente di testare in modo coerente e riproducibile gli scenari di guasto in un ambiente controllato, rendendolo adatto per l'integrazione automatizzata delle pipeline e i test di regressione.

Al contrario, l'ingegneria del caos non è deterministica. In altre parole, si basa su ipotesi e verifica il modello mentale dell'utente in base al modo in cui l'applicazione e le sue dipendenze (persone, processi e tecnologia) assorbono, si adattano e alla fine si riprendono da situazioni di fallimento impreviste. Pertanto, l'ingegneria del caos si concentra sulla end-to-end verifica di modalità di guasto sconosciute, con l'obiettivo di individuare tempestivamente i difetti e correggerli prima che si trasformino in eventi su larga scala. L'ingegneria del caos favorisce l'apprendimento continuo e dovrebbe essere praticata attraverso una pipeline separata o esperimenti ad hoc che consentano di eseguire più esperimenti in qualsiasi momento senza ostacolare la produttività dello sviluppatore nell'implementazione del codice.



Il processo di ingegneria del caos spesso inizia con una caos game day, ossia un evento dedicato in cui i team inseriscono intenzionalmente errori o guasti controllati nelle proprie applicazioni. La giornata di gioco è progressiva: inizia in ambienti di livello inferiore (come lo sviluppo o il test) e passa gradualmente ad ambienti di livello superiore (come la messa in scena e la pre-produzione) man mano che aumenta la fiducia. Spostandosi sistematicamente in questi ambienti, i team possono verificare che i propri sistemi tollerino correttamente i guasti iniettati prima di entrare in produzione. Questa progressione metodica garantisce che, nel momento in cui vengono condotti esperimenti sul caos negli ambienti di produzione, i team abbiano acquisito una notevole fiducia nelle capacità di resilienza del sistema. Il processo del game day è un approccio proattivo per identificare punti deboli e vulnerabilità nell'architettura e nelle pratiche operative di un'applicazione, eliminando al contempo lo stress dell'apprendimento durante un'interruzione imprevista della produzione.

Il valore dell'ingegneria del caos

I sistemi complessi sono onnipresenti nel mondo di oggi. Svolgono un ruolo fondamentale in molti aspetti della nostra vita, dai mercati finanziari all'assistenza sanitaria. Ci aspettiamo che questi sistemi siano sempre operativi. Tuttavia, i sistemi complessi sono spesso vulnerabili a eventi e comportamenti imprevisti che possono avere conseguenze significative. Le organizzazioni devono pianificare le interruzioni invece di chiedersi se si verificheranno. Possono farlo applicando test di scenario a tutti i loro servizi aziendali critici o mission-critical. È qui che entra in gioco l'ingegneria del caos.

L'ingegneria del caos offre un approccio alla gestione di sistemi complessi che può aiutare a mitigare i rischi e migliorare la resilienza. Il processo di preparazione agli esperimenti sul caos richiede ai team di sviluppare ipotesi sul comportamento del sistema, il che approfondisce la comprensione di come i sistemi sono costruiti e di come funzionano. Questa preparazione spesso rivela lacune mentali, approfondimenti architettonici e conoscenze operative che altrimenti potrebbero rimanere sconosciute. Promuovendo la comprensione di come i sistemi complessi reagiscono ai guasti,

l'ingegneria del caos promuove una maggiore trasparenza e responsabilità nella progettazione e nella gestione dei sistemi. Quanto più spesso l'organizzazione pratica l'ingegneria del caos, tanto meglio si prepara a operare. L'ingegneria del caos consente di stabilire le migliori pratiche per la progettazione di applicazioni resilienti in grado di resistere ai guasti dei componenti con un impatto minimo o nullo sull'utente. Ciò garantisce che le applicazioni critiche operino entro i livelli di servizio e la tolleranza all'impatto previsti, migliorando al contempo la conoscenza dei sistemi da parte del team.

Prepararsi a condizioni avverse

Quando si sviluppa AWS, si utilizzano diversi tipi di servizi, tra cui servizi zonali come Amazon Elastic Compute Cloud (Amazon EC2), servizi regionali come Amazon Simple Storage Service (Amazon S3), servizi globali come (IAM), servizi software AWS Identity and Access Management as a service (SaaS) di terze parti e servizi locali. Ogni tipo di servizio espone diversi domini di errore di cui devi tenere conto. Come ci si prepara agli eventi autoinflitti o agli eventi causati da terze parti su cui l'organizzazione non ha alcun controllo?

Per capire in che modo l'applicazione potrebbe rispondere a condizioni avverse, puoi usare AWS Fault Injection Service ().AWS FIS AWS FIS è un servizio completamente gestito per l'esecuzione di esperimenti di iniezione dei guasti in modo controllato. È possibile utilizzare questo servizio per inserire scenari AWS forniti, ad esempio interruzioni dell'alimentazione nella zona di disponibilità e problemi di connettività tra regioni, oppure creare esperimenti personalizzati concatenando un'ampia gamma di azioni di guasto fornite dal servizio. AWS FIS consente ai vostri team di esercitarsi e imparare continuamente in che modo la loro applicazione reagirebbe ai guasti più comuni e correggerebbe i difetti non appena li rilevano.

Praticare l'ingegneria del caos controllato

I principi chiave degli esperimenti sul caos controllato sono:

- Inizia con un ambiente simile al tuo ambiente di produzione.
- Stabilisci un'ipotesi e interrompi le condizioni per il tuo esperimento.
- Inizia in piccolo.
- Esercita il controllo sui tuoi esperimenti sul caos.
- Imposta l'ambito dell'impatto.
- Conosci la linea di base del tuo servizio.
- Pianifica gli esperimenti.

- Prima correggi e poi sperimenta.
- Monitora attentamente l'esperimento.
- Impara dai tuoi risultati.
- Dai priorità ai risultati, correggi e verifica.
- Diffondi le conoscenze in tutta l'organizzazione.

Per scalare con successo l'ingegneria del caos, è necessario implementare esperimenti sul caos in modo controllato. Quando lo usi AWS FIS, puoi creare condizioni di arresto utilizzando gli CloudWatchallarmi Amazon. Puoi incorporare queste condizioni in un modello di esperimento per assicurarti che gli esperimenti vengano interrotti se superano i limiti e ripristinati all'ultimo stato noto. AWS FIS fornisce anche leve di sicurezza. Quando attivi queste leve, AWS FIS interrompe e ripristina tutti gli esperimenti in corso nell'account in the Regione AWS, compresi gli esperimenti con più account, e impedisce l'avvio di nuovi esperimenti. In questo modo si evita l'insorgenza di errori durante determinati periodi di tempo, ad esempio durante gli orari di negoziazione, gli eventi di vendita o il lancio di prodotti, o in risposta agli allarmi sullo stato dell'applicazione. La leva di sicurezza rimane innestata finché non viene disinnestata manualmente.

Quando si conduce un esperimento basato sul caos, è necessario definire misure di protezione per prevenire effetti collaterali indesiderati sull'ambiente, soprattutto se esiste la possibilità che l'esperimento influisca sulle applicazioni in produzione. Quando pianificate l'esperimento, anticipate gli eventuali effetti negativi che potrebbe avere su altre applicazioni dell'ambiente. Ad esempio, altre applicazioni potrebbero ricevere messaggi errati dall'applicazione che fa parte dell'esperimento, registrare elevati volumi di richieste o riscontrare un conflitto di risorse se condividono l'infrastruttura. Documenta questi rischi e risolvi eventuali problemi noti o inaccettabili prima di eseguire l'esperimento.

Guida introduttiva all'ingegneria del caos

Prima di condurre un esperimento, ti consigliamo di mettere in atto alcuni elementi essenziali per sfruttare al meglio le tue pratiche di ingegneria del caos. Questi elementi essenziali includono:

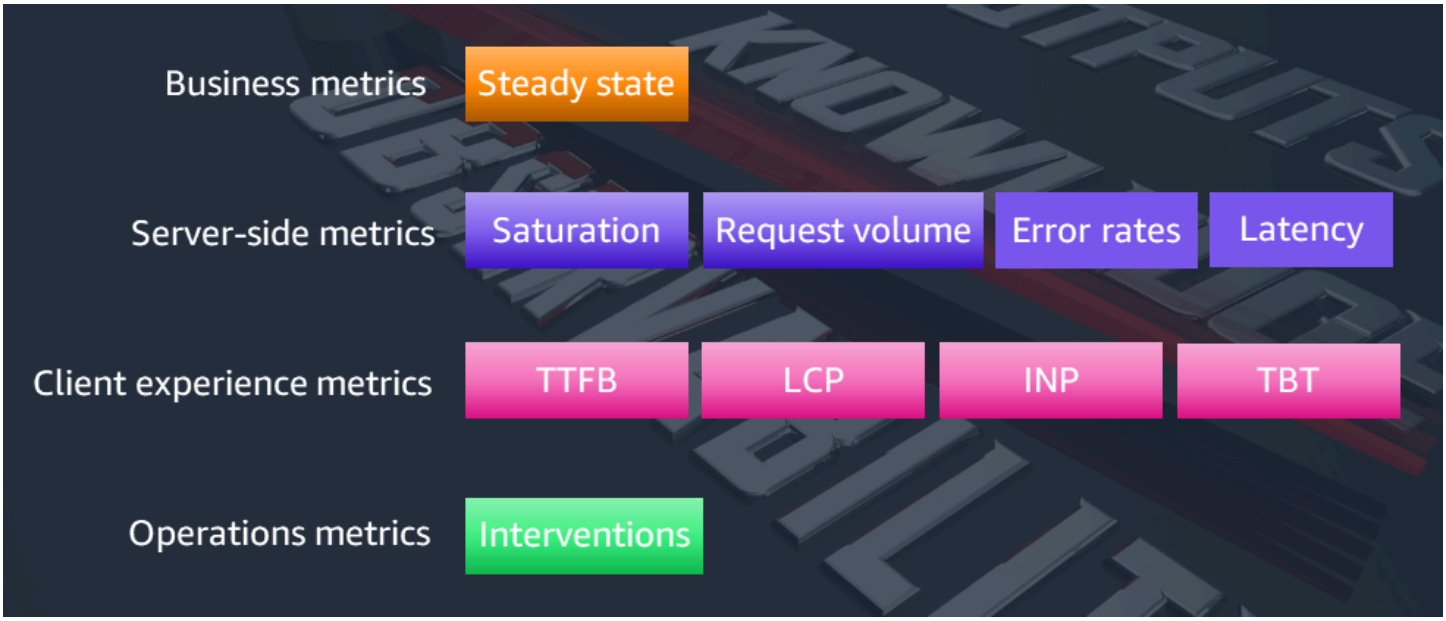
- Osservabilità (metriche, registrazione, tracciamento delle richieste)
- Un elenco di eventi o guasti del mondo reale che vorresti esplorare
- Sponsorizzazione della resilienza organizzativa tramite il consenso della leadership
- Assegnazione di priorità ai risultati critici, in base al potenziale impatto aziendale, rispetto alle nuove funzionalità che vengono scoperte durante gli esperimenti sul caos

Osservabilità per esperimenti sul caos

L'osservabilità, che comprende metriche, registrazione e tracciamento delle richieste, svolge un ruolo chiave nell'ingegneria del caos. Quando esegui un esperimento, vorrai comprendere le metriche aziendali, le metriche lato server, le metriche dell'esperienza del cliente e le metriche operative. Senza osservabilità, non sarete in grado di definire il comportamento allo stato stazionario o creare un esperimento significativo per verificare se le vostre ipotesi sull'applicazione sono vere.

Metriche

Il diagramma seguente mostra i tipi di metriche che è possibile monitorare per esperimenti di caos per diversi tipi di applicazioni.



- Metriche aziendali: lo stato stazionario indica il normale funzionamento del sistema ed è definito dalle metriche aziendali. Può essere rappresentato da transazioni al secondo (TPS), flussi di clic al secondo, ordini al secondo o una misurazione simile. L'applicazione mostra uno stato stazionario quando funziona come previsto. Pertanto, verificate che l'applicazione sia integra prima di eseguire gli esperimenti. Lo stato stazionario non significa necessariamente che non vi sarà alcun impatto sull'applicazione quando si verifica un errore, poiché una percentuale di errori potrebbe rientrare nei limiti accettabili. Lo stato stazionario è la linea di base. Ad esempio, lo stato stazionario di un sistema di pagamenti potrebbe essere definito come l'elaborazione di 300 TPS con una percentuale di successo del 99 percento e un tempo di andata e ritorno di 500 ms. Visivamente, pensate allo stato stazionario come a un elettrocardiogramma (ECG). Se lo stato stazionario del sistema cambia improvvisamente, sai che c'è un problema con il tuo servizio.
- Metriche lato server: per comprendere le prestazioni delle risorse durante l'esperimento, è necessario approfondire le loro prestazioni prima, durante e dopo l'esperimento. Per misurare l'impatto delle tue risorse AWS, puoi usare Amazon CloudWatch. CloudWatch è un servizio che monitora le applicazioni, risponde ai cambiamenti delle prestazioni, ottimizza l'uso delle risorse e fornisce informazioni sullo stato operativo. Durante i tuoi esperimenti, ti consigliamo di acquisire metriche lato server come saturazione, volumi di richieste, tassi di errore e latenza.
- Metriche sull'esperienza del cliente: attivo AWS, puoi acquisire metriche utente reali utilizzando CloudWatchRUM per simulare le richieste degli utenti tramite strumenti come Locust, Grafana k6, Selenium o Puppeteer. Le metriche relative agli utenti reali sono fondamentali per le organizzazioni che conducono esperimenti di ingegneria del caos. Monitorando l'impatto sugli utenti reali durante un esperimento, i team possono ottenere un quadro preciso di come i guasti e le interruzioni

influiranno sui clienti durante la produzione. Le metriche chiave relative all'esperienza del cliente sono Time to First Byte (TTFB), Largest Contentful Paint (LCP), Interaction to Next Paint (INP) e Total Blocking Time (TBT).

- Metriche operative: gli interventi misurano l'efficacia della mitigazione degli errori in modo automatizzato, ad esempio la corretta latenza delle richieste dei client durante il riavvio di pod, attività o istanze EC2 con meccanismi come il controller di replica o la scalabilità automatica. La possibilità di intervenire automaticamente durante un guasto è direttamente correlata a una buona esperienza utente. Capire se nel tempo questi meccanismi di mitigazione subiscano variazioni è fondamentale. Definendo le metriche per le mitigazioni automatiche riuscite e fallite, si creano linee guida che aiutano a identificare potenziali regressioni in tutto il sistema.

Registrazione dei log

La registrazione centralizzata è fondamentale per comprendere lo stato dei componenti dell'applicazione prima, durante e dopo un esperimento basato sul caos. Ti consigliamo di raccogliere i log di tutti i componenti dell'applicazione per creare una visione consolidata di ciò che stava facendo ogni componente al momento dell'iniezione dell'esperimento. Ciò fornisce un quadro chiaro del flusso dell' end-to-endesperimento.

Tracciamento delle richieste

Il tracciamento delle richieste consente di osservare il flusso di ogni singola richiesta tra i componenti dell'applicazione per ottenere una comprensione completa dell'impatto che l'errore iniettato ha sul sistema e sulle sue dipendenze. Tracciando le richieste, è possibile vedere come l'errore si propaga attraverso diversi servizi e componenti, in modo da poter valutare meglio la portata dell'interruzione. Per tracciare le tue richieste AWS, puoi usare. AWS X-Ray

Scenari di fallimento da iniettare nel caos: esperimenti

L'obiettivo dell'inserimento dei guasti più comuni nell'applicazione è capire come l'applicazione reagisce a questi eventi imprevisti, in modo da poter creare meccanismi di mitigazione e rendere il sistema resiliente a tali errori. Inoltre, è consigliabile utilizzare l'ingegneria del caos per riprodurre gli scenari di errore cronologici per verificare che i meccanismi di mitigazione funzionino ancora come previsto e non abbiano subito variazioni nel tempo.

Considerate i seguenti eventi quando pianificate i vostri esperimenti di ingegneria del caos.

Modalità di guasto	Description
Compromissione del server	Riavvia le istanze EC2, elimina i pod Kubernete s o le attività di Amazon Elastic Container Service (Amazon ECS) per capire come reagisce l'applicazione a tali arresti anomali.
errori API	Inserisci i guasti nel tuo servizio per comprende re il comportamento dell'applicazione. AWS APIs
Problemi di rete	Introduci latenza o congestione o blocca le connessioni per simulare i problemi di rete del mondo reale.
AWS Compromissione della zona di disponibi lità	Riproduci la compromissione di un'intera zona di disponibilità per verificare il ripristino tra le zone.
Regione AWS compromissione della connettiv ità	Riproduci un problema di rete Regioni AWS per verificare in che modo le risorse della regione secondaria reagiscono a tale evento.
Errori del database	Esegui il failover delle repliche del database o danneggia i dati o rendi irraggiungibili le istanze del database, per comprendere l'impatto sulle applicazioni e sulle strategie di ripristino.
Sospensione della replica del database e di Amazon S3	Metti in pausa la replica del database o di Amazon Simple Storage Service (Amazon S3) tra le zone di disponibilità o Regioni AWS per comprendere l'impatto delle applicazioni a valle.
Degrado dello storage	Metti in pausa l'I/O, rimuovi i volumi Amazon Elastic Block Store (Amazon EBS) o elimina i file per verificare la durabilità e il ripristino dei dati.

Modalità di guasto	Description
Compromissione della dipendenza	Riduci o riduci le prestazioni dei servizi downstream o upstream da cui dipendi, compresi i servizi di terze parti, per comprende rne il end-to-end flusso e l'impatto sui tuoi clienti.
Sbalzi di traffico	Genera picchi nel traffico degli utenti per testare le funzionalità di scalabilità automatica e scopri come il tempo di avvio a freddo potrebbe influire sullo stato generale dell'applicazione.
Esaurimento delle risorse	Massimizza CPU, memoria e spazio su disco per verificare il corretto degrado dell'appl icazione.
Guasti a cascata	Avvia guasti primari che si ripercuotono in cascata su applicazioni e componenti a valle.
Implementazioni errate	Implementa modifiche o configurazioni problematiche per verificare i meccanismi di rollback.

Sponsorizzazione della resilienza organizzativa

L'ingegneria del caos offre il massimo valore quando viene applicata in tutta l'organizzazione. Ti consigliamo di collaborare con uno sponsor esecutivo che possa aiutarti a stabilire obiettivi di resilienza all'interno dell'organizzazione, a rimuovere la paura, l'incertezza e i dubbi sul dominio e ad avviare il processo di trasformazione per rendere la resilienza una responsabilità di tutti.

Per sostenere l'idea aziendale di creare uno studio di ingegneria del caos, associa le attività di ingegneria del caos ai tuoi progetti aziendali critici. Fare della resilienza una risorsa e un fattore di accelerazione vi aiuterà a monitorare il successo nel tempo. Inizia con un conteggio degli incidenti critici al mese o al trimestre, il tempo medio di ripristino e l'impatto che questi incidenti hanno causato sui clienti e sull'organizzazione. Stabilisci con i tuoi team l'obiettivo di ridurre il numero di incidenti in

un periodo da 6 a 12 mesi, grazie ai miglioramenti apportati a tutti gli stack di applicazioni in risposta agli esperimenti di ingegneria del caos.

Misura se gli incidenti sono altamente ripetitivi. Ad esempio, supponiamo che un certificato TLS scaduto porti a tempi di inattività perché i client non riescono a stabilire una connessione affidabile. Se si verificano più incidenti in un anno a causa di più scadenze di certificati TLS, puoi eseguire un esperimento sulla scadenza di un certificato TLS e verificare che i tuoi team ricevano avvisi o siano in grado di mitigare automaticamente tali problemi. Ciò contribuirà a garantire la resilienza a tali errori.

Per tenere traccia dei progressi dell'ingegneria del caos nel tempo, acquisisci le seguenti metriche per evidenziare il valore dell'ingegneria del caos durante il ciclo di vita di un'applicazione:

- Tasso di incidenti ridotto: monitora il numero di incidenti di produzione nel tempo e correla questo numero con l'adozione dell'ingegneria del caos. Si prevede che il tasso di incidenti gravi diminuirà.
- Tempo medio di risoluzione (MTTR) migliorato: calcola il tempo medio necessario per risolvere gli incidenti e monitora questi dati per vedere se, grazie all'ingegneria del caos, migliorano nel tempo.
- Maggiore disponibilità delle applicazioni: utilizza le metriche a livello di servizio per mostrare i miglioramenti della disponibilità man mano che aumenta la resilienza delle applicazioni attraverso esperimenti di caos.
- Tempi di commercializzazione più rapidi: Chaos Engineering può fornire la sicurezza necessaria per lanciare offerte innovative più rapidamente, perché sapete che le vostre applicazioni sono resilienti. Tieni traccia degli aumenti della velocità di rilascio dei prodotti.
- Riduzione dei costi operativi: quantifica se indicatori come il rumore di allarme, il carico operativo e lo sforzo manuale per gestire le applicazioni diminuiscono con l'adozione di pratiche basate sul caos.
- Incremento della fiducia: intervistate sviluppatori, ingegneri dell'affidabilità del sito (SREs) e altro personale tecnico per valutare se l'ingegneria del caos abbia rafforzato la loro fiducia nella resilienza delle applicazioni. Le percezioni contano.
- Esperienze clienti migliorate – Connect Chaos Engineering a risultati positivi per i clienti, come un minor numero di interruzioni del servizio, rollback e interruzioni.

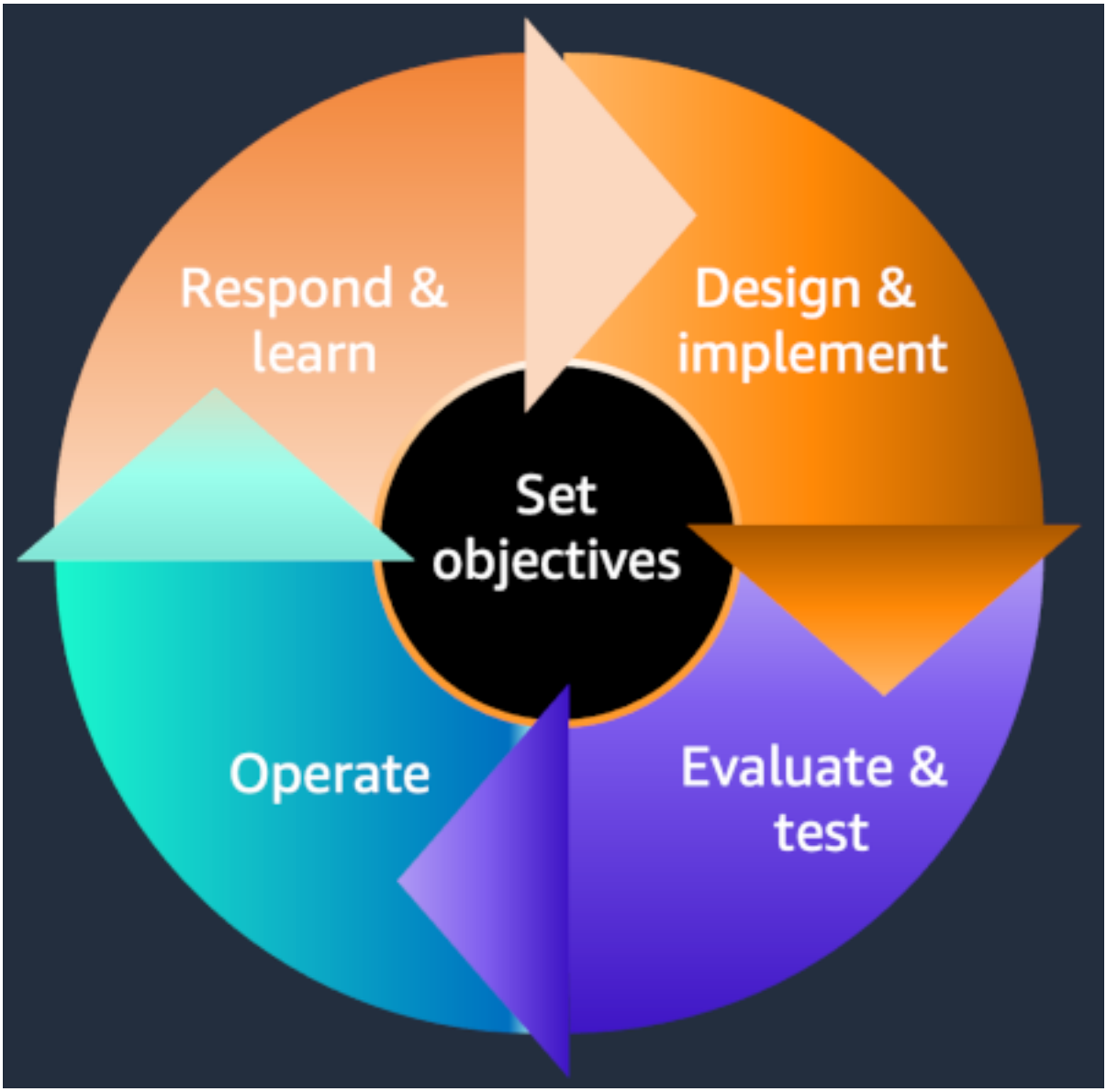
Dare priorità alla riparazione

Quando si eseguono esperimenti di caos, è probabile che si identifichino aree di miglioramento in cui l'applicazione non funziona come previsto. La correzione di tali elementi diventerà parte integrante del vostro backlog, a cui dovrete dare priorità insieme ad altre attività, come lo sviluppo

di funzionalità. Ti consigliamo di dedicare del tempo a questi miglioramenti per evitare futuri errori. Valuta la possibilità di dare priorità a queste attività di apprendimento e correzione in base al livello di impatto che potrebbero causare. I risultati che hanno un impatto diretto sulla resilienza o sulla sicurezza dell'applicazione dovrebbero avere la priorità sulle nuove funzionalità, per evitare l'impatto sui clienti. Se il team ha difficoltà a dare priorità alle azioni correttive rispetto allo sviluppo delle funzionalità, prendi in considerazione la possibilità di contattare il tuo sponsor esecutivo per garantire che le priorità siano stabilite in base alla tolleranza al rischio aziendale.

Implementazione del caos engineering su AWS

L'ingegneria del caos fa parte della fase di valutazione e test del ciclo di vita della AWS resilienza, come illustrato nel diagramma seguente. Le applicazioni distribuite non funzionano in modo isolato da altre applicazioni o client, quindi consigliamo di esaminare l'intero ciclo di vita della resilienza. Il cambiamento è costante per le applicazioni distribuite man mano che la rete si evolve, le applicazioni upstream e downstream subiscono cambiamenti e l'utilizzo dei client cambia nel tempo.



Per capire in che modo queste modifiche all'applicazione potrebbero influire sulla sua resilienza, includi l'ingegneria del caos nelle tue operazioni. day-to-day Puoi implementare esperimenti sul caos in diversi modi:

- Ad hoc: puoi eseguire esperimenti sul caos come esperimenti singoli per risolvere un problema o una domanda specifici.
- Chaos game days: si tratta di eventi strutturati e ricorrenti progettati per verificare l'affidabilità e la resilienza di un'applicazione. Lo scopo di un Chaos Game Day è identificare potenziali problemi o carenze di resilienza tra persone, processi e tecnologie e mettere in pratica i processi e le procedure per identificare, mitigare e rispondere agli incidenti.
- Chaos pipeline: integrazione continua e distribuzione continua (CI/CD) is about building new features and deploying them safely throughout the environments. To implement chaos engineering experiments, create a chaos pipeline that's separate from your CI/CD pipeline. To understand why, let's assume that you want to add a single chaos engineering experiment to your CI/CDpipeline che inietta una crescente perdita di pacchetti per i componenti a valle). L'esperimento viene eseguito 3 volte e richiede 5 minuti per essere completato ogni volta. La perdita di pacchetti aumenta dal 10% al 20% al 30% ad ogni esecuzione e il completamento dell'esperimento richiede complessivamente 15 minuti. Se disponi di 100 implementazioni parallele, dovrai attendere 1500 minuti per il completamento di un singolo esperimento. Se dovessi eseguire 10 esperimenti, l'impatto sugli sviluppatori sarebbe insopportabile. Su larga scala, l'ingegneria del caos necessita di una propria pipeline che consenta di eseguire esperimenti parallelamente al processo del ciclo di vita dello sviluppo del software (SDLC).
- Implementazioni Canarie: le isole Canarie forniscono un ambiente di test per esperimenti sul caos. Indirizzando una piccola percentuale del traffico verso un servizio Canary o utilizzando metodi come il mirroring o il replay del traffico, è possibile verificare nuove infrastrutture o modifiche al codice senza impatto sul sistema di produzione stabile. Potete eseguire esperimenti contro il canarino e iniettare eventuali errori, se necessario, in modo da limitare la portata dell'impatto sull'utente finale.
- Esperimenti pianificati: è possibile pianificare esperimenti per verificare i meccanismi di ripristino prevedibili per l'applicazione. Usa esperimenti pianificati per riprodurre eventi noti di frequente per comprendere in che modo i tuoi sistemi possono riprendersi da eventi come la chiusura di un' EC2 istanza basata su un gruppo di scalabilità automatico, la rimozione di un pod Kubernetes o l'eliminazione di un'attività Amazon ECS.

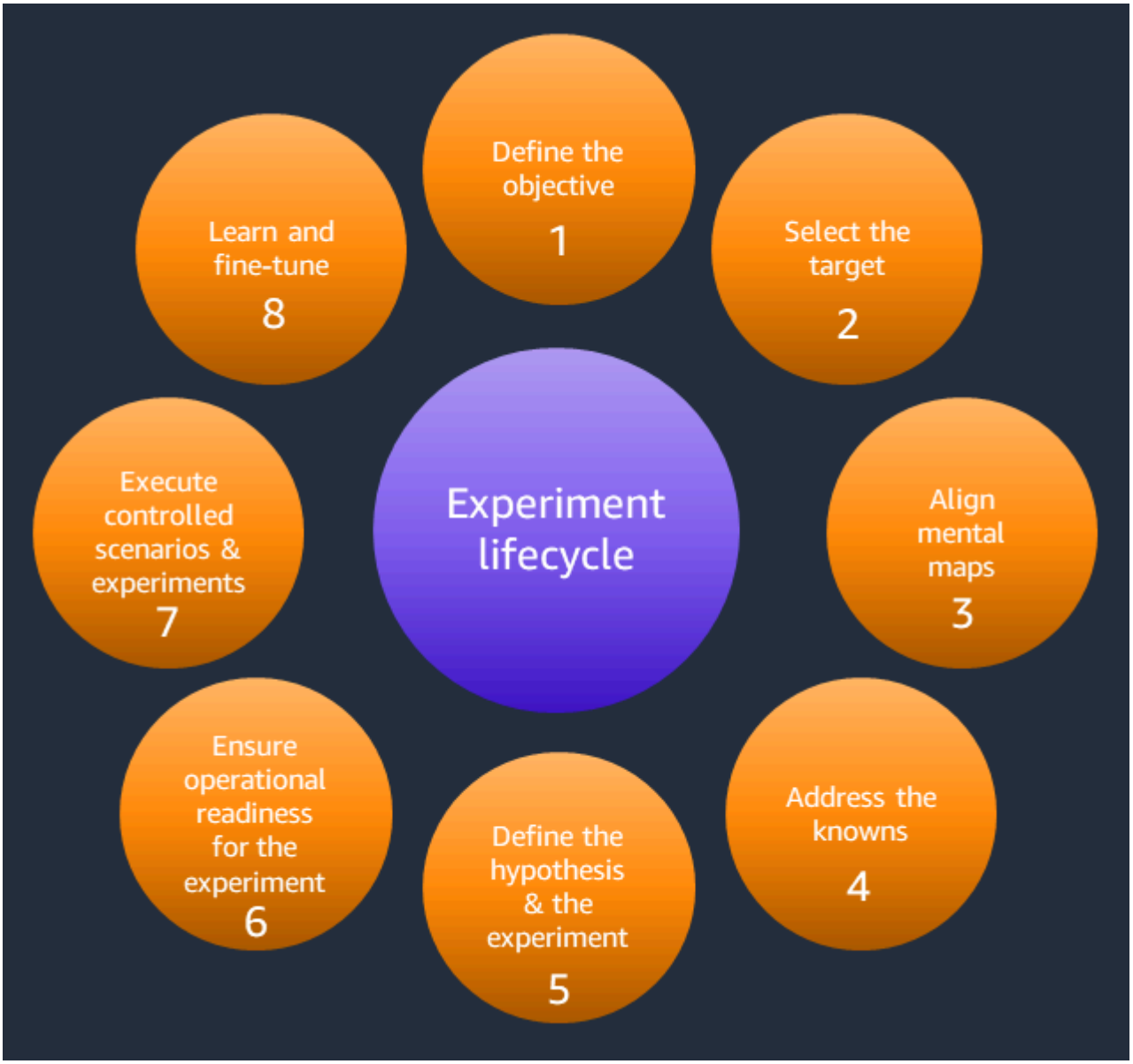
Ciclo di vita dell'esperimento di ingegneria del caos continuo

Come discusso nella sezione precedente, è possibile implementare esperimenti di ingegneria del caos in diversi modi. In tutti i casi, la chiave per creare un esperimento di caos efficace è comprendere l'applicazione, gli incidenti storici e le soluzioni implementate, nonché comprendere chiaramente le aree su cui indagare, come la resilienza o la sicurezza. Le conoscenze acquisite sull'applicazione consentono di formulare un'ipotesi sui potenziali punti deboli dell'applicazione e di comprendere in che modo l'applicazione rileverà, correggerà e ripristinerà l'errore.

Il ciclo di vita del Chaos Experiment include le seguenti fasi:

1. Definisci l'obiettivo dell'esperimento.
2. Seleziona l'applicazione di destinazione.
3. Allinea le mappe mentali.
4. Risolvi i problemi noti con la tua applicazione.
5. Definisci l'ipotesi e l'esperimento.
6. Garantire la prontezza operativa per l'esperimento.
7. Esegui scenari ed esperimenti controllati.
8. Impara e perfeziona l'esperimento.

Questi passaggi sono illustrati nel diagramma e discussi nelle sezioni seguenti.



Definisci gli obiettivi e stabilisci le aspettative

Prima di ogni esperimento, assicurati che i tuoi obiettivi e le tue aspettative siano specifici, misurabili, raggiungibili, pertinenti e limitati nel tempo. Definisci chiaramente quanto segue:

- Identifica potenziali guasti o punti deboli nei sistemi e nei servizi, per capire come potrebbero influire sugli utenti. Ciò include l'identificazione di possibili modalità di errore, come arresti anomali del server, errori di rete o bug del software, e la valutazione del loro potenziale impatto sulle prestazioni e sull'affidabilità complessive del sistema.

- Quantifica l'impatto dei guasti definendo i principali indicatori di rischio (KRIs) sui tuoi sistemi e servizi. Ciò include la misurazione dell'effetto dei guasti quando metriche come latenza, velocità effettiva e tassi di errore si discostano dal loro stato stazionario. Comprendendo l'impatto di tali deviazioni, è possibile dare priorità agli sforzi per mitigare i guasti in base ai rischi aziendali.
- Sviluppa e verifica strategie per mitigare o prevenire i guasti. Ciò include l'identificazione di potenziali soluzioni, come la ridondanza, la correzione degli errori o le strategie di fallback, e la verifica della loro efficacia in un ambiente controllato. Verificando queste strategie, potete assicurarvi di essere efficaci nel prevenire o mitigare i guasti e di poterle implementare nei vostri sistemi di produzione con sicurezza.
- Migliora la risposta agli incidenti e i processi di disaster recovery. Riproducendo gli errori in un ambiente controllato, è possibile testare i processi di risposta agli incidenti, identificare potenziali colli di bottiglia o lacune e perfezionare le procedure di disaster recovery. Ciò consente di essere pronti a rispondere in modo rapido ed efficace in caso di guasti imprevisti.

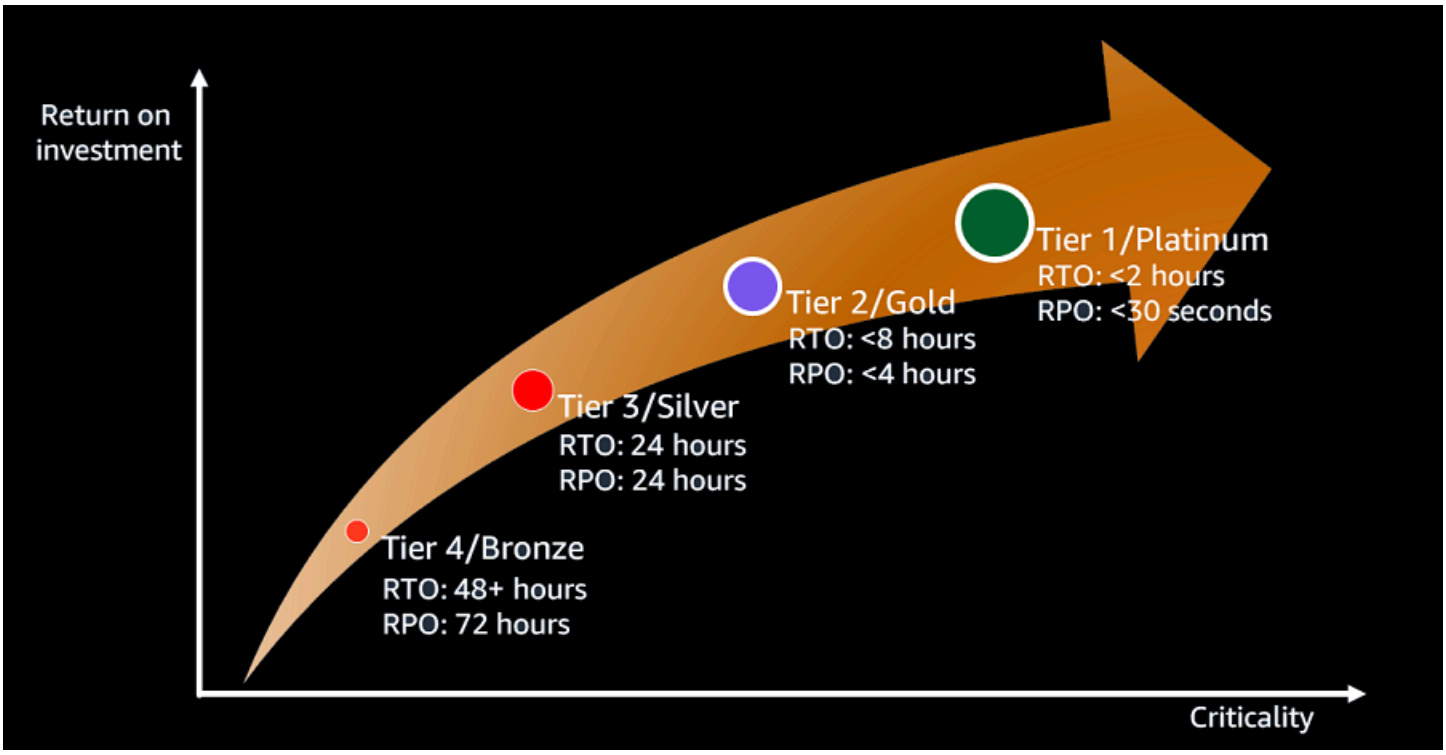
Seleziona l'applicazione di destinazione

L'ingegneria del caos è una tecnica potente, ma richiede un'attenta definizione delle priorità per massimizzare il valore. Quando decidete dove concentrare le vostre attività di ingegneria del caos, iniziate a considerare i servizi essenziali della vostra azienda. Chiedete ai vostri team di ripetere le fasi del ciclo di vita dello sviluppo del software e iniziate innanzitutto a inserire errori negli ambienti di test. Le applicazioni aziendali critiche sono direttamente legate ai ricavi, all'esperienza del cliente e alle operazioni principali. Gli esperimenti sul caos condotti su questi servizi possono scoprire vulnerabilità che possono avere gravi ripercussioni sull'organizzazione, e potenzialmente su interi mercati, se non vengono risolte. Ad esempio, concentrati innanzitutto sui servizi rivolti ai clienti, come i sistemi di trading o i sistemi di ordinazione. Dare priorità a questi servizi centrali offre la massima protezione per ogni investimento di tempo.

Dopo i servizi critici, esamina i componenti fondamentali come database, code di messaggi, reti e servizi condivisi. APIs Questi potrebbero essere usati come componenti o servizi condivisi in tutta l'organizzazione, quindi il loro fallimento causerebbe problemi diffusi. La conferma della resilienza dei servizi di infrastruttura offre la certezza che non comprometteranno le applicazioni dipendenti che li sovrastano. Ad esempio, un esperimento di ingegneria del caos che smonta un cluster Kafka rivela molto sulla tolleranza ai guasti delle applicazioni downstream. Sebbene l'infrastruttura di sistema non sia direttamente rivolta ai clienti, è un obiettivo primario dell'ingegneria del caos.

Non dimenticate di mappare le lacune mentali relative a persone, processi, informazioni sulle strutture e dipendenze da terze parti, poiché queste possono causare gravi interruzioni se non sono in linea con gli obiettivi di tolleranza all'impatto dell'organizzazione. Per ulteriori informazioni sulla misurazione del ROI dell'ingegneria del caos, leggi Quantificare il ROI dell'ingegneria del caos nel documento strategico Investire nell'ingegneria del caos come necessità strategica.

Il diagramma seguente mostra il ritorno sull'investimento derivante dall'esecuzione di esperimenti sul caos su diversi livelli di servizi.



Allinea le mappe mentali (scoperta delle applicazioni)

Quando esegui esperimenti ad hoc o giornate di gioco, inizierai il processo di scoperta dell'applicazione organizzando una sessione sulla lavagna incentrata sulla mappatura dei dettagli dell'applicazione. (Se esegui gli esperimenti nella pipeline del caos, avrai già allineato quella mappa mentale definendo l'applicazione di destinazione.) Un buon approccio per comprendere le lacune mentali consiste nel chiedere al membro del team più giovane di disegnare prima un diagramma della domanda, e poi chiedere ai membri dello staff più senior di aggiungerlo progressivamente al diagramma. Questo permetterà di scoprire eventuali lacune di comprensione tra i diversi livelli di esperienza.

Il diagramma dovrebbe illustrare sia le dipendenze dirette a monte che quelle a valle dell'applicazione, nonché eventuali integrazioni critiche di terze parti. Assicurati che vi sia un allineamento sul flusso previsto di una richiesta tramite l'applicazione. Mappa i flussi di lavoro e i percorsi degli utenti chiave per avere più chiarezza su come i clienti utilizzano l'applicazione. Prendi in considerazione l'utilizzo di un diagramma di sequenza per acquisire queste informazioni.

Dopo questa sessione collaborativa, il team dovrebbe disporre di un modello mentale condiviso dell'applicazione, delle sue dipendenze critiche e delle sue capacità di monitoraggio, nonché di una comprensione dei rischi necessari per prendere una decisione informata se procedere o annullare un potenziale esperimento di caos.

Risolvete i problemi noti dell'applicazione

Gli esperimenti di Chaos Engineering sono progettati per far emergere in modo proattivo i difetti di un'applicazione. Inserendo errori come aumenti della latenza, riavvii del server o interruzioni dell'alimentazione della zona di disponibilità, è possibile verificare la capacità dell'applicazione di tollerare interruzioni realistiche. Tuttavia, questo processo presuppone un livello di stabilità e integrità di base nell'applicazione di destinazione. Eseguire esperimenti di caos su un'applicazione già problematica rischia di mascherare problemi più profondi.

Prima di intraprendere l'ingegneria del caos, i team devono risolvere eventuali difetti, bug e problemi di prestazioni noti nella loro applicazione.

Definisci l'ipotesi e l'esperimento

Gli incidenti passati che hanno causato interruzioni dell'applicazione o di altre applicazioni all'interno dell'organizzazione possono costituire un'ottima fonte di idee per sperimentare il caos. Ad esempio, le interruzioni precedenti sono state causate da errori di configurazione o dalla mancanza di modelli di resilienza? Rivedere le cronologie degli incidenti e ripercorrere le cause profonde di questi fallimenti nel mondo reale attraverso esperimenti sul caos è un modo efficace per sviluppare la resilienza contro problemi simili in futuro.

Un'altra preziosa fonte di concetti sperimentali può provenire direttamente dagli ingegneri, dagli architetti e dagli operatori che hanno più familiarità con un'applicazione. Consentire ai membri del team di presentare ipotetici scenari di errore che ritengono possano compromettere in modo significativo l'applicazione consente di raccogliere idee basate su conoscenze privilegiate. Il team addetto all'applicazione può quindi valutare quale di questi scenari proposti potrebbe avere il

maggiore impatto potenziale o esporre i maggiori rischi sconosciuti. Intraprendere esperimenti di caos mirati a scenari così rischiosi e meno compresi può generare importanti insegnamenti e prevenire problemi futuri.

Una terza fonte di idee proviene dall'esecuzione di modelli di resilienza per anticipare le condizioni che porterebbero a perdite aziendali identificate. Alcuni esercizi di modellizzazione della resilienza hanno un approccio basato sui componenti per la creazione di un modello di resilienza, mentre altri hanno un approccio basato sui sistemi. Un approccio basato sui componenti pone la domanda: «Cosa succede quando il componente x è sottoposto a un carico estremo o è guasto?» Il team che sviluppa il modello di resilienza ipotizza quindi l'effetto di tale scenario sull'applicazione più ampia e identifica i controlli di monitoraggio e prevenzione attualmente in atto per rilevare e mitigare gli effetti dello scenario. In alternativa, un approccio basato sui sistemi segue un processo dall'alto verso il basso per evidenziare uno stato indesiderato dell'applicazione, ad esempio «La vetrina web mostra livelli di inventario obsoleti», e invita il team dell'applicazione a prevedere quale condizione o condizioni indurrebbero l'applicazione a comportarsi in questo modo.

Garantite la prontezza operativa per l'esperimento

Sono necessari indicatori quantificabili per misurare l'impatto delle condizioni avverse sull'applicazione e sul suo comportamento, come descritto in precedenza nella sezione sull'osservabilità. La possibilità di misurare il comportamento dell'applicazione consente di determinare se le condizioni avverse hanno influito sull'applicazione e in che misura.

Il modo migliore per capire se c'è un impatto sull'applicazione è misurarne lo stato stazionario. Lo stato stazionario misura l'aspetto del normale funzionamento e in genere si allinea agli indicatori di esperienza aziendale e del cliente per una determinata applicazione. Prima di passare alla fase successiva, assicuratevi di disporre dell'osservabilità necessaria per comprendere l'impatto e di disporre dei meccanismi di rollback nel caso in cui l'esperimento non si risolva come previsto.

Esegui esperimenti e scenari controllati

Noi AWS sconsigliamo di condurre un esperimento iniziale basato sul caos su un'applicazione in produzione. Lo scopo di un esperimento basato sul caos è imparare qualcosa di nuovo sul comportamento dell'applicazione in condizioni di stress. Il comportamento dell'applicazione potrebbe essere imprevedibile durante l'esperimento, quindi eseguire un esperimento per la prima volta in produzione potrebbe avere conseguenze sul cliente. Pertanto, dovreste sempre eseguire un esperimento iniziale basato sul caos in un ambiente di livello inferiore con un potenziale minimo di

impatto sugli utenti del mondo reale, e quindi ripetere l'analisi degli ambienti dopo aver verificato e aver avuto la certezza che l'applicazione sia in grado di assorbire, adattarsi e riprendersi dalle azioni iniettate.

Pianificate accuratamente ogni esperimento utilizzando un documento che riporti i dettagli chiave, simile al documento di pianificazione dell'esperimento fornito nell'appendice. Alcuni dei campi critici da includere sono la definizione dello stato stazionario, l'ipotesi e il metodo di iniezione in caso di guasto. La pianificazione, l'esecuzione e l'analisi di un esperimento di caos possono essere incluse in un unico artefatto.

Dopo aver finalizzato il piano scritto per l'esperimento, preparate il codice necessario per inserire le interruzioni pianificate descritte nel documento.

Per cogliere il potenziale impatto durante l'esperimento, assicuratevi che siano presenti meccanismi di osservabilità. Se non disponete ancora di un metodo automatizzato per acquisire i risultati degli esperimenti, ad esempio i report degli AWS FIS esperimenti, identificate i membri del team che prenderanno appunti durante l'esecuzione, acquisite schermate delle dashboard e guidate il team durante l'esperimento.

Impara e perfeziona

Dopo ogni esperimento, riunitevi in squadra per rivedere e riflettere sull'esperimento del caos. Sforzatevi consapevolmente di mantenere una mentalità irreprensibile. Il vostro obiettivo dovrebbe essere quello di instaurare un dialogo aperto e costruttivo che si concentri interamente sulla massimizzazione dell'apprendimento e non sull'attribuzione di colpe.

Inizia esaminando la definizione e l'ipotesi dello stato stazionario per l'esperimento. L'applicazione si è comportata come previsto? Ci sono state sorprese che hanno invalidato le ipotesi? Discutete le osservazioni su come l'applicazione ha reagito durante l'esperimento, sia positive che negative. I dati raccolti (metriche, registri, schermate e così via) dovrebbero raccontare esattamente cosa è successo.

Affrontate questa revisione dei dati con curiosità anziché con giudizio e individuate le aree in cui è possibile apportare miglioramenti alla progettazione delle applicazioni, alla documentazione, al monitoraggio o ad altre funzionalità basate sulle conoscenze acquisite. Queste azioni vengono raccolte come progetti successivi per rendere l'applicazione più resiliente.

Grazie a questo approccio irreprensibile, puoi avere conversazioni sincere su cosa è andato storto e su come risolverlo. Presumete l'intenzione positiva di tutte le persone coinvolte e abbiate fiducia nel

fatto che stessero lavorando per ottenere buoni risultati. Il vostro obiettivo condiviso è la crescita e la progressione organizzativa attraverso l'apprendimento e l'adattamento continui. Le revisioni degli esperimenti Chaos, condotte in modo costruttivo e irreprensibile, offrono al team uno spazio sicuro in cui acquisire informazioni preziose che rendono le applicazioni e l'organizzazione più affidabili e resilienti a lungo termine. L'attenzione rimane sugli apprendimenti, non sulle persone. Per diffondere le conoscenze tra i tuoi team, pubblica il rapporto sui risultati dell'esperimento in una posizione centrale e pubblicizza i risultati in modo che altri possano trarne vantaggio.

Scalare l'ingegneria del caos in tutta l'organizzazione

Man mano che l'organizzazione adotta l'ingegneria del caos, la sua standardizzazione e implementazione presenteranno delle sfide. Nelle fasi iniziali della maturità, è probabile che team diversi utilizzino strumenti e varianti diversi del processo di ingegneria del caos descritto nelle sezioni precedenti. Allo stesso tempo, alcuni team potrebbero non dare priorità o adottare affatto l'ingegneria del caos, nonostante i suoi potenziali vantaggi. Le sezioni seguenti forniscono indicazioni su come superare queste sfide.

Nel complesso, il vostro approccio all'ingegneria del caos dovrebbe essere progettato per trovare un equilibrio tra leadership centralizzata e partecipazione decentralizzata. Questo equilibrio aiuta a garantire che l'ingegneria del caos sia integrata nel processo di sviluppo e che le conoscenze acquisite siano condivise all'interno dell'organizzazione.

Istituire una pratica di ingegneria del caos

La standardizzazione della pratica dell'ingegneria del caos può accelerarne l'adozione. La condivisione degli insegnamenti tratti dagli esperimenti tra i team può aumentare il ritorno sugli investimenti nell'ingegneria del caos.

Costruisci un centro di eccellenza centralizzato o riunisci un gruppo di esperti in materia, come parte della tua pratica di ingegneria del caos. Essendo una piccola funzione centralizzata, questo team può operare tra team di sviluppo software, infrastruttura, sicurezza e business e mantenere gli standard utilizzati da tali team. Per semplicità, il centro di eccellenza è denominato team di pratica centralizzato e i gruppi che applicano l'ingegneria del caos sono chiamati team di pratica nel resto di questa guida.

Ruolo del team di studio centralizzato

Il team di studio centralizzato è responsabile dello sviluppo e dell'implementazione di pratiche di ingegneria del caos in tutta l'organizzazione. Lavorano a stretto contatto con i team di pratica per guidarli nella progettazione e nella conduzione di esperimenti e garantire che gli esperimenti siano preziosi per l'azienda. Il team di pratica centralizzato fornisce inoltre indicazioni e supporto ai team di sviluppo, infrastruttura e sicurezza per aiutarli a integrare l'ingegneria del caos nei loro processi di sviluppo.

Le responsabilità principali di un team di studio centralizzato di ingegneria del caos includono quanto segue:

- Abilitazione: una funzione centralizzata di ingegneria del caos funge da facilitatore per introdurre la pratica dell'ingegneria del caos attraverso giornate di gioco e workshop. Guidano i team nel processo di ingegneria del caos, inclusa la selezione degli scenari di fallimento, la definizione di ipotesi e la produzione di report da condividere con l'intera organizzazione. Il team di studio centralizzato dovrebbe possedere i materiali di formazione e lavorare per migliorare le competenze dei team di pratica nell'uso dell'ingegneria del caos.
- Consulenza: il team di pratica centralizzato può anche svolgere un ruolo consultivo per supervisionare gli esperimenti condotti dai team di pratica. La loro esperienza e conoscenza possono garantire che gli esperimenti apportino valore all'azienda e siano condotti in modo sicuro. Allo stesso modo, il team può supervisionare l'esecuzione e il resoconto di un esperimento per guidare chi è alle prime armi nell'ingegneria del caos.
- Marketing e monitoraggio del valore: comunicare il valore aziendale dell'ingegneria del caos è fondamentale per il successo di un programma di questo tipo. Ogni team che partecipa a esperimenti di ingegneria del caos deve raccogliere dati dagli esperimenti condotti in tutta l'azienda e dimostrare il valore dell'investimento dell'organizzazione nell'ingegneria del caos. Ciò include la quantificazione e la celebrazione del numero di incidenti evitati durante ogni esperimento, dei tempi di inattività che si sarebbero verificati se l'esperimento fosse fallito e dell'impatto complessivo sull'azienda se gli scenari di fallimento si fossero verificati durante la produzione. Raccogliendo e centralizzando tali dati provenienti da tutti i team e rendendoli disponibili in tutta l'organizzazione, il team addetto allo studio centralizzato può tracciare e influenzare il valore derivante dall'adozione dell'ingegneria del caos in tutta l'organizzazione.
- Standard: il team di pratica centralizzato dovrebbe gestire e gestire il processo di conduzione degli esperimenti sul caos, i modelli per la pianificazione e la rendicontazione degli esperimenti e gli strumenti utilizzati per condurre gli esperimenti.

  Il team centrale dovrebbe possedere e gestire i modelli di pianificazione degli esperimenti, i modelli di report sugli esperimenti, la documentazione dei processi e i materiali di abilitazione. La documentazione sulle migliori pratiche e i materiali di abilitazione forniscono indicazioni ai team esperti su argomenti quali i guardrail da utilizzare per limitare l'impatto di un esperimento, quando condurre un esperimento in produzione e come far evolvere nel tempo l'uso dell'ingegneria del caos. Per esempi di modelli e output, consulta l'appendice.

  Il team di studio centralizzato dovrebbe inoltre essere responsabile del processo di conduzione di un esperimento, comprese le comunicazioni e l'escalation, e quando e come comunicare con gli altri team dell'organizzazione prima o durante un esperimento. Il processo dovrebbe inoltre indicare quando sono necessari dei guardrail.

Il team di pratica centralizzato dovrebbe inoltre selezionare e possedere gli strumenti principali per condurre esperimenti sul caos (ad esempio, strumenti come). AWS FIS La scelta e l'implementazione di strumenti supplementari, come gli strumenti per la generazione del carico, dovrebbero essere lasciate alla decisione dei team di pratica. I team di professionisti dovrebbero essere in grado di adattare il processo e gli strumenti complessivi per soddisfare al meglio le proprie esigenze.

Ruolo delle squadre di allenamento

Il team centralizzato è responsabile della definizione della strategia generale di ingegneria del caos, mentre i team addetti alla pratica partecipano al processo e sono i responsabili dello sviluppo e dell'esecuzione degli esperimenti. Ciò contribuisce a garantire che gli esperimenti siano pertinenti a ogni prodotto o servizio specifico e che le conoscenze acquisite siano utilizzabili e possano essere applicate per migliorare l'affidabilità e la resilienza del prodotto. Il team di studio centralizzato funge da mentore e proprietario degli standard e dei processi di ingegneria del caos dell'organizzazione. Tuttavia, per evitare che il team centralizzato diventi un collo di bottiglia, i singoli team di allenamento dovranno imparare dalla pratica centrale per eseguire autonomamente esperimenti sul caos.

Creazione di una comunità di pratica

Oltre a creare un team centralizzato, ti consigliamo di creare una comunità informale di professionisti interessati all'ingegneria del caos. Questa community offre una piattaforma per condividere conoscenze, best practice ed esperienze tra i team di pratica e l'organizzazione in generale.

La comunità di pratica può essere gestita dal team di studio centralizzato di Chaos Engineering, ma chiunque all'interno dell'organizzazione può diventare membro della comunità. Il team centralizzato può sfruttare la comunità di pratica per trasmettere aggiornamenti e reperire informazioni utili e raccogliere feedback dai team di pratica che utilizzano gli standard e i processi gestiti dal team centralizzato. La community fungerà da circuito di feedback per informare il team centralizzato sull'efficacia delle pratiche di ingegneria del caos tra i team di pratica. Il team di studio centralizzato può quindi modificare la documentazione e gli elementi di supporto per supportare al meglio i team di prodotto.

Incorporare l'ingegneria del caos nella resilienza operativa

Un esperimento basato sul caos è un investimento da parte dell'azienda per prevenire incidenti durante la produzione. Sarà necessario determinare dove l'azienda può ottenere il massimo ritorno su questo investimento. L'organizzazione può collaborare con il team dello studio di ingegneria del caos centralizzato per aggiornare i propri standard e determinare quali prodotti sono sufficientemente critici da richiedere la sperimentazione del caos.

Processo di sviluppo dei sistemi

L'ingegneria del caos e gli esperimenti sul caos devono essere eseguiti ripetutamente come parte del ciclo di vita di un'applicazione. Allo stesso modo in cui i team eseguono regolarmente i test di disaster recovery, dovrebbero condurre esperimenti sul caos e giornate di gioco in modo continuativo e periodico durante tutto l'anno. Questo approccio migliora il modo in cui un'organizzazione prevede, osserva e risponde agli incidenti.

Conclusioni

La disciplina dell'ingegneria del caos ha fatto molta strada nell'ultimo decennio. È stata adottata in diversi settori e ha aiutato le organizzazioni a creare servizi resilienti e aumentare la soddisfazione dei clienti. (Per esempi di come le organizzazioni hanno implementato queste pratiche, consulta Chaos Engineering Stories.) L'ingegneria del caos consente alle organizzazioni di ridurre i rischi per le loro applicazioni mission-critical iniettando guasti controllati a tutti i livelli dello stack applicativo, compresi i servizi dei provider cloud. La capacità di influire su un intero stack di applicazioni in modo controllato consente resilienza continua, miglioramento dell'eccellenza operativa, osservabilità e architettura orientata al ripristino senza lo stress delle interruzioni della produzione. Questo approccio porta anche a migliori pratiche di test di resilienza in tutta l'organizzazione. Per iniziare ad abbracciare l'ingegneria del caos, organizza una giornata o un workshop dedicato ai giochi del caos per mostrare il valore che l'ingegneria del caos può offrire alla tua organizzazione.

Risorse

AWS FIS implementazioni di modelli di fallimento:

- AWS Fault Injection Service Da utilizzare per dimostrare la resilienza delle applicazioni multiregione e Multi-AZ (post sul blog)AWS
- AWS Fault Injection Simulator supporta esperimenti di ingegneria del caos su Amazon EKS Pods (AWS post sul blog)
- Annuncio delle nuove funzionalità di AWS Fault Injection Simulator per i carichi di lavoro Amazon ECS (post sul blog)AWS
- Migliora la resilienza delle applicazioni con i parametri di volume di Amazon EBS e AWS Fault Injection Simulator (post sul blog)AWS
- Ingegneria del caos che sfrutta AWS Fault Injection Simulator in un ambiente multi-account (post sul blog) AWS AWS
- Esperimenti di caos su Amazon RDS con AWS Fault Injection Simulator (AWS post sul blog)
- Creazione di applicazioni serverless resilienti utilizzando l'ingegneria del caos (post sul blog)AWS
- Usa FIS per interrompere un'istanza spot (workshop)AWS
- Automatizzazione dell'ingegneria del caos nelle vostre pipeline di distribuzione (post della community)AWS

Altre risorse:

- Investire nell'ingegneria del caos come necessità strategica
- Storie di ingegneria del caos
- CloudWatchDocumentazione Amazon
- Documentazione Amazon CloudWatch RUM
- Documentazione di AWS X-Ray
- Documentazione di AWS FIS

Appendice: Documenti di esempio

I documenti di esempio forniti in questa sezione utilizzano un sito fittizio per l'adozione di animali domestici (PetSite) come applicazione di destinazione per un esperimento caotico.

- Documento di pianificazione dell'esperimento
- Documento sui risultati dell'esperimento

Documento di pianificazione dell'esperimento

Stato stazionario

Process name (Nome del processo)	Sito di adozione di animali domestici
Architettura fisica	(Collegamento al diagramma dell'architettura.)
Architettura logica	(Collegamento al diagramma logico.)
Definire lo stato stazionario	Il tempo medio di caricamento della pagina, misurato utilizzando Largest Contentful Paint (LCP), per il sito di adozione di animali domestici è pari o inferiore a 2,5 secondi con una latenza del 99 percentile (P99) di 4,0 secondi o inferiore con una linea di base di 5000 utenti simultanei.
Metriche dello stato stazionario	Metrica LCP acquisita sulla base di utenti e metriche preferenziali (latenza, velocità effettiva , tassi di errore, saturazione).
Osservabilità in stato stazionario	L'LCP verrà acquisito dal browser dell'utente, inviato ad Amazon CloudWatch e ispeziona to con CloudWatch RUM. In un periodo di 60 secondi, la media e il tempo LCP P99 verranno aggregati per tutte le richieste in quel periodo.

Process name (Nome del processo)	Sito di adozione di animali domestici
	Le metriche auree di primo livello vengono acquisite utilizzando. CloudWatch
Processo per raggiungere lo stato stazionario	Grafana K6 verrà utilizzato per creare un carico che simula i normali livelli di traffico di produzione di circa 5.000 utenti simultanei.

Requisiti di osservabilità

I team dovrebbero essere in grado di visualizzare quanto segue:

- Stato stazionario: cosa verrà osservato per verificare che l'applicazione sia in condizioni normali?
- Condizione di guasto: come verrà visualizzata la condizione di guasto nella dashboard? Ad esempio:
  - Allarmi che devono essere attivati
  - Registri che dovrebbero essere generati
- Impatto sul guasto: cosa occorre osservare per visualizzare i componenti che si prevede subiranno un impatto (ambito dell'impatto)?
- Recupero: come verrà visualizzato e misurato il ripristino per acquisire l'MTTR?
- Debug: risoluzione dei problemi relativi agli errori degli esperimenti.

La tabella seguente fornisce suggerimenti ed esempi per una tabella dei requisiti di osservabilità. È necessario definire cosa osservare in base al proprio esperimento specifico.

Cosa deve essere osservato	Link allo strumento di osservabilità	Cosa viene osservato
Fonte di input	Cruscotto Grafana K6	• Numero di container in esecuzione • Richieste al secondo

Cosa deve essere osservato	Link allo strumento di osservabilità	Cosa viene osservato
Stato generale delle applicazioni	• CloudWatch Pannello sull'adozione di animali domestici • Dashboard sull'esperienza utente sull'adozione di animali domestici (RUM)	• Numero di nodi valido di Amazon EKS • Utilizzo della CPU del nodo Amazon EKS
Stato del flusso di lavoro	CloudWatch Pannello sull'adozione di animali domestici	Tempo LCP, metriche privilegiate
Tracce	Dashboard X-Ray per l'adozione di animali domestici	• Richiedi la latenza • Request Count (Numero richieste) • Numero di fallimenti
Logs	CloudWatch Registri di adozione di animali domestici	Eventuali errori riscontrati dai pod verranno inviati a CloudWatch Logs.

Definizione dell'esperimento

Nome dell'esperimento	Stress della CPU del PetSite pod Amazon EKS
Codice sorgente dell'esperimento	(Link al repository dei sorgenti dell'esperimento.)
Descrizione dell'esperimento	Questo esperimento analizza come un aumento dell'utilizzo della CPU dell' PetSite application pod influirebbe sull'esperienza complessiva del cliente. Iniettando lo stress

Nome dell'esperimento	Stress della CPU del PetSite pod Amazon EKS
	della CPU in ogni PetSite pod funzionante, saremo in grado di capire se c'è un impatto sui clienti e l'entità di tale impatto.
Requisiti o parametri dell'esperimento	Carico dell'applicazione: media di produzione Selettore di etichette per pod: `app=petsite`
Durata esperimento	10 minuti
Ambiente	Ambiente di test Alpha
Risorse mirate agli esperimenti	PetSite contenitori per applicazioni
Linea di base dell'esperimento introdotta tramite lo strumento di generazione del carico	• Il 54% delle richieste ha un LCP inferiore a 2,5 secondi. • Il 46% delle richieste ha un LCP inferiore a 4 secondi. • Non vengono osservati errori.
Condizione di backoff	Nessuno

Ipotesi

Cosa succede se	Impatto	Ripristino
Cosa accadrebbe allo stato stazionario se i pod delle PetSite applicazioni registrassero o causassero un utilizzo della CPU superiore al 60% per 10 minuti in condizioni di traffico normale a livello di produzione?	I tempi LCP rimarranno inferiori a 2,5 secondi per il 50% degli utenti, con P99 pari o inferiore a 4,0 secondi. Il consumatore dovrebbe essere in grado di caricare la PetSite pagina di destinazione.	Rilevamento: • Lo stress della CPU verrà rilevato dagli allarmi configurati in CloudWatch. • Le metriche LCP genereranno anche allarmi relativi al

Cosa succede se	Impatto	Ripristino
		peggioramento dell'esperienza utente. Autoriparazione: • La natura distribuita dell'architettura dei microservizi significa che molte istanze di pod sono in esecuzione su più zone di disponibilità. • Il piano di controllo del cluster EKS sposterà il traffico lontano dai pod interessati e lancerà nuovi pod sui nodi di lavoro. Recupero: Quando l'utilizzo della CPU torna alla normalità, l'LCP dovrebbe ripristinarsi automaticamente.

Processo sperimentale

Adatta il seguente step-by-step processo di esempio al tuo esperimento specifico:

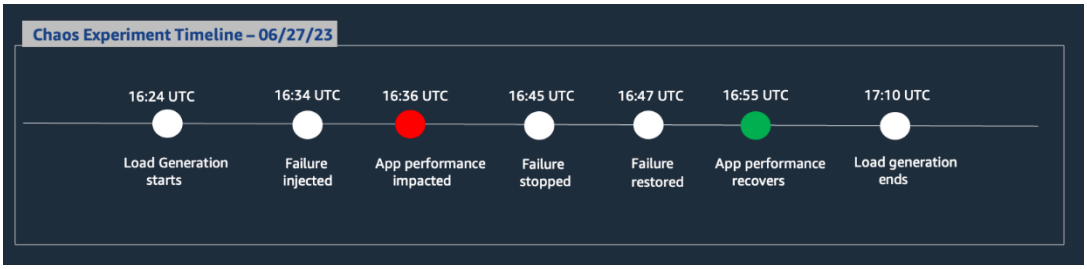
1. Convalida l'accesso e la funzionalità di tutti gli Amazon CloudWatch, CloudWatch RUM e i AWS X-Ray dashboard.
2. Convalida lo stato dell'ambiente applicativo:
   a. Verifica che il cluster EKS sia integro utilizzando la CloudWatch dashboard.

b. Visita l'implementazione dell'applicazione Test Pet Adoption Site all'URL di esempio.

3. Avvia un carico per raggiungere lo stato stazionario:
   a. Verificate che il generatore di carico sia in funzione e invii 5000 richieste al secondo.
   b. Attendi 5 minuti affinché l'applicazione raggiunga lo stato stazionario.
   c. Conferma lo stato stazionario dell'applicazione utilizzando la dashboard CloudWatch RUM.
4. Avvia un guasto (esperimento):
   a. Apri la AWS FIS console.
   b. Esegui l' pet-adoption-pod-stressesperimento.
   c. Conferma che l'esperimento sia in esecuzione nella console.
5. Osserva l'impatto dell'errore sulla tua applicazione:
   a. Acquisisci schermate dal CloudWatch RUM e dai CloudWatch dashboard e annota eventuali punti dati anomali.
   b. Una volta completato l'esperimento AWS FIS, acquisisci schermate aggiuntive per registrare se l'applicazione torna allo stato stazionario in assenza di stress e annota eventuali anomalie nei punti dati.
   c. Se lo stato stazionario non riprende, procedi per ripristinare l'applicazione e registra i passaggi eseguiti.
6. Verifica che l'ambiente sia tornato alla normalità:
   - Esamina tutte le metriche aziendali, relative all'esperienza utente, alle applicazioni e all'infrastruttura per verificare che il sistema sia tornato a uno stato noto. Se utile, acquisisci schermate della dashboard.

Cronologia dell'esperimento

Assicurati di registrare la sequenza temporale dell' end-to-endesperimento, iniziando con la generazione del carico, l'iniezione del guasto, l'osservazione dell'impatto e il ripristino dell'applicazione, e terminando quando interrompi la generazione del carico. Questo è illustrato nell'esempio seguente.



Risultati dell'esperimento

ID di esecuzione dell'esperimento	Risultati dell'esperimento
`PET-ADOPT-EXP-23`	(Link ai risultati dell'esperimento.)

Difetti identificati

- Il cluster Kubernetes non ha rilevato danni alla CPU dei PetSite pod, quindi non ha pianificato implementazioni aggiuntive.
- Non vi è stato alcun aumento dei tassi di errore 4XX o 5XX come risultato di questo esperimento.
- Dobbiamo modificare il controllo dello stato del pod per tenere conto dell'impatto sull'LCP in caso di limitazioni di risorse.

Documento sui risultati dell'esperimento

Configurazione

Documenta le configurazioni specifiche dell'esperimento. Ad esempio:

- Generazione del carico impostata per simulare 5.000 utenti che emettono un totale di 85 richieste al secondo.

Prerequisiti

- È stato verificato che il sito di adozione degli animali domestici funzionasse nell'ambiente alpha test.

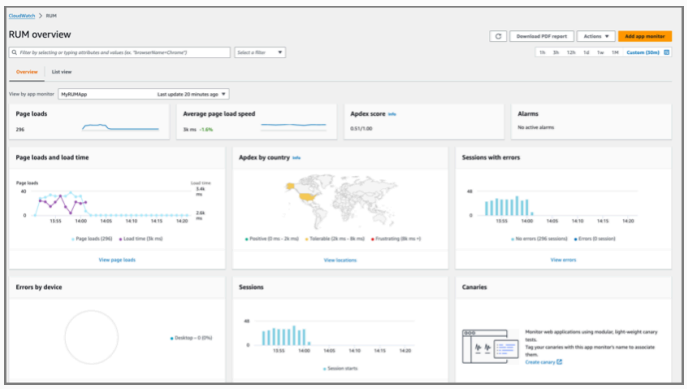
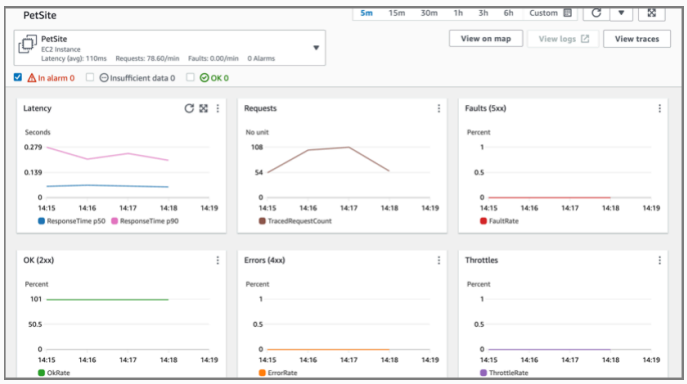
- È stato verificato che il modello di esperimento fosse configurato per applicare lo stress della CPU ai pod delle PetSite applicazioni in esecuzione nel cluster EKS. I pod delle applicazioni sono stati identificati dall'etichetta Kubernetes. `app=petsite`
- È stato confermato che Load è in esecuzione e genera 85 richieste al secondo.

Stato stazionario

Documenta le misure adottate per raggiungere lo stato stazionario e come lo hai verificato. Ad esempio:

Per l'implementazione di test del sito di adozione di animali domestici, viene generato un carico di 85 RPS per simulare lo stato stazionario. Il CloudWatch RUM e i CloudWatch dashboard sono stati esaminati per verificare che tutte le metriche aziendali e applicative rientrassero negli intervalli normali precedenti all'esecuzione dell'esperimento.

Dati di osservabilità:

Expected (Atteso)	Osservato
• LCP è inferiore a 4 secondi per P99 di richieste. • La latenza di risposta è inferiore a 500 ms. • Non ci sono errori 4XX o 5XX.	 

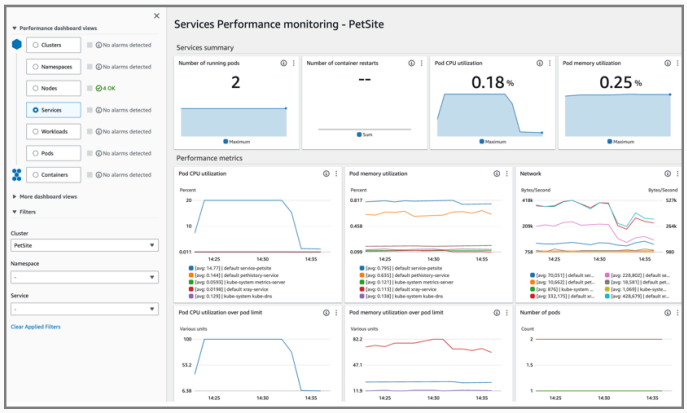
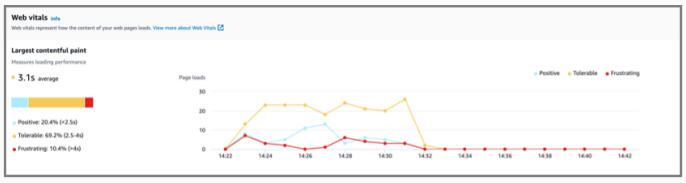
Iniezione per errore

AWS FIS è stato utilizzato per iniettare errori utilizzando il modello di esperimento (fornire un collegamento). L'esperimento era impostato per durare 10 minuti ed era stato configurato un rollback se i nodi di lavoro subivano uno stress della CPU superiore al 60 percento.

Osservazione dei guasti

Il CloudWatch RUM e i CloudWatch dashboard sono stati esaminati per tenere traccia dello stato stazionario dell'applicazione (definito utilizzando le metriche LCP). Le schermate sono state acquisite nella tabella seguente.

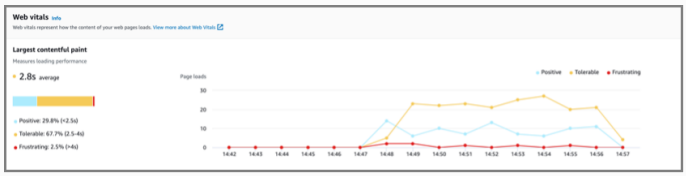
Dati di osservabilità:

Expected (Atteso)	Osservato
• L'LCP dovrebbe rimanere inferiore a 4 secondi per P99. • Il tempo di risposta deve rimanere inferiore a 500 ms. • Non dovrebbero verificarsi errori 4XX o 5XX.	 

Ripristino

Dopo che lo stress è stato rimosso (l' AWS FIS esperimento è stato completato e lo stress della CPU è stato rimosso dai pod), l'applicazione dovrebbe riprendere il suo normale stato stazionario. Non dovrebbe essere richiesto alcun intervento manuale.

Dati di osservabilità:

Expected (Atteso)	Osservato (screenshot)
LCP P99 dovrebbe essere inferiore a 4 secondi con una media inferiore a 2,5 secondi.	

Cronologia dei documenti

La tabella seguente descrive le modifiche significative apportate a questa guida. Per ricevere notifiche sugli aggiornamenti futuri, puoi abbonarti a un feed RSS.

Modifica	Descrizione	Data
Pubblicazione iniziale	—	4 aprile 2025

AWS Glossario delle linee guida prescrittive

I seguenti sono termini di uso comune nelle strategie, nelle guide e nei modelli forniti da AWS Prescriptive Guidance. Per suggerire voci, utilizza il link Fornisci feedback alla fine del glossario.

Numeri

7 R

Sette strategie di migrazione comuni per trasferire le applicazioni sul cloud. Queste strategie si basano sulle 5 R identificate da Gartner nel 2011 e sono le seguenti:

- Refactor/re-architect — Sposta un'applicazione e modificane l'architettura sfruttando appieno le funzionalità native del cloud per migliorare l'agilità, le prestazioni e la scalabilità. Ciò comporta in genere la portabilità del sistema operativo e del database. Esempio: migra il tuo database Oracle locale all'edizione Amazon PostgreSQL-Compatible Aurora.
- Ridefinire la piattaforma (lift and reshape): trasferisci un'applicazione nel cloud e introduci un certo livello di ottimizzazione per sfruttare le funzionalità del cloud. Esempio: migra il tuo database Oracle locale ad Amazon Relational Database Service (Amazon RDS) per Oracle in. Cloud AWS
- Riacquistare (drop and shop): passa a un prodotto diverso, in genere effettuando la transizione da una licenza tradizionale a un modello SaaS. Esempio: migra il tuo sistema di gestione delle relazioni con i clienti (CRM) su. Salesforce.com
- Eseguire il rehosting (lift and shift): trasferisci un'applicazione sul cloud senza apportare modifiche per sfruttare le funzionalità del cloud. Esempio: migra il tuo database Oracle locale su Oracle su un'istanza EC2 in. Cloud AWS
- Trasferire (eseguire il rehosting a livello hypervisor): trasferisci l'infrastruttura sul cloud senza acquistare nuovo hardware, riscrivere le applicazioni o modificare le operazioni esistenti. Esegui la migrazione dei server da una piattaforma locale a un servizio cloud per la stessa piattaforma. Esempio: migra un'applicazione suMicrosoft Hyper-V. AWS
- Riesaminare (mantenere): mantieni le applicazioni nell'ambiente di origine. Queste potrebbero includere applicazioni che richiedono una rifattorizzazione significativa che desideri rimandare a un momento successivo e applicazioni legacy che desideri mantenere, perché non vi è alcuna giustificazione aziendale per effettuarne la migrazione.
- Ritirare: disattiva o rimuovi le applicazioni che non sono più necessarie nell'ambiente di origine.

A

A2A () Agent-to-Agent

Un protocollo statico per la collaborazione tra agenti che supporta la delega delle attività e il trasferimento dello stato.

ABAC

Vedi controllo degli accessi basato sugli attributi.

servizi astratti

Vedi servizi gestiti.

ACIDO

Vedi atomicità, consistenza, isolamento, durata.

migrazione attiva-attiva

Un metodo di migrazione del database in cui i database di origine e di destinazione vengono mantenuti sincronizzati (utilizzando uno strumento di replica bidirezionale o operazioni di doppia scrittura) ed entrambi i database gestiscono le transazioni provenienti dalle applicazioni di connessione durante la migrazione. Questo metodo supporta la migrazione in piccoli batch controllati anziché richiedere una conversione una tantum. È più flessibile ma richiede più lavoro rispetto alla migrazione attiva-passiva.

migrazione attiva-passiva

Un metodo di migrazione del database in cui i database di origine e di destinazione vengono mantenuti sincronizzati, ma solo il database di origine gestisce le transazioni provenienti dalle applicazioni di connessione mentre i dati vengono replicati nel database di destinazione. Il database di destinazione non accetta alcuna transazione durante la migrazione.

Agente

Un sistema di intelligenza artificiale in grado di ragionare, pianificare e intraprendere azioni in modo autonomo utilizzando strumenti per raggiungere gli obiettivi.

Agente Ops

Pratiche operative per la creazione, il test, l'implementazione e l'esecuzione di agenti di intelligenza artificiale in produzione su larga scala.

funzione aggregata

Una funzione SQL che opera su un gruppo di righe e calcola un singolo valore restituito per il gruppo. Esempi di funzioni aggregate includono `SUM` e. `MAX`

Intelligenza artificiale

Vedi intelligenza artificiale.

AIOps

Guarda le operazioni di intelligenza artificiale.

anonimizzazione

Il processo di eliminazione permanente delle informazioni personali in un set di dati. L'anonimizzazione può aiutare a proteggere la privacy personale. I dati anonimi non sono più considerati dati personali.

anti-modello

Una soluzione utilizzata frequentemente per un problema ricorrente in cui la soluzione è controproducente, inefficace o meno efficace di un'alternativa.

controllo delle applicazioni

Un approccio alla sicurezza che consente l'uso solo di applicazioni approvate per proteggere un sistema dal malware.

portfolio di applicazioni

Una raccolta di informazioni dettagliate su ogni applicazione utilizzata da un'organizzazione, compresi i costi di creazione e manutenzione dell'applicazione e il relativo valore aziendale. Queste informazioni sono fondamentali per il processo di scoperta e analisi del portfolio e aiutano a identificare e ad assegnare la priorità alle applicazioni da migrare, modernizzare e ottimizzare.

intelligenza artificiale (IA)

Il campo dell'informatica dedicato all'uso delle tecnologie informatiche per svolgere funzioni cognitive tipicamente associate agli esseri umani, come l'apprendimento, la risoluzione di problemi e il riconoscimento di schemi. Per ulteriori informazioni, consulta la sezione Che cos'è l'intelligenza artificiale?

operazioni di intelligenza artificiale (AIOps)

Il processo di utilizzo delle tecniche di machine learning per risolvere problemi operativi, ridurre gli incidenti operativi e l'intervento umano e aumentare la qualità del servizio. Per ulteriori

informazioni su come viene utilizzato AIOps nella strategia di migrazione AWS , consulta la guida all'integrazione delle operazioni.

crittografia asimmetrica

Un algoritmo di crittografia che utilizza una coppia di chiavi, una chiave pubblica per la crittografia e una chiave privata per la decrittografia. Puoi condividere la chiave pubblica perché non viene utilizzata per la decrittografia, ma l'accesso alla chiave privata deve essere altamente limitato.

atomicità, consistenza, isolamento, durabilità (ACID)

Un insieme di proprietà del software che garantiscono la validità dei dati e l'affidabilità operativa di un database, anche in caso di errori, interruzioni di corrente o altri problemi.

Controllo degli accessi basato su attributi (ABAC)

La pratica di creare autorizzazioni dettagliate basate su attributi utente, come reparto, ruolo professionale e nome del team. Per ulteriori informazioni, consulta ABAC for AWS nella documentazione AWS Identity and Access Management (IAM).

fonte di dati autorevole

Una posizione in cui è archiviata la versione principale dei dati, considerata la fonte di informazioni più affidabile. È possibile copiare i dati dalla fonte di dati autorevole in altre posizioni allo scopo di elaborarli o modificarli, ad esempio anonimizzandoli, oscurandoli o pseudonimizzandoli.

Zona di disponibilità

Una posizione distinta all'interno di un edificio Regione AWS che è isolata dai guasti in altre zone di disponibilità e offre una connettività di rete economica e a bassa latenza verso altre zone di disponibilità nella stessa regione.

AWS Cloud Adoption Framework (CAF)AWS

Un framework di linee guida e best practice AWS per aiutare le organizzazioni a sviluppare un piano efficiente ed efficace per passare con successo al cloud. AWS CAF organizza le linee guida in sei aree di interesse chiamate prospettive: business, persone, governance, piattaforma, sicurezza e operazioni. Le prospettive relative ad azienda, persone e governance si concentrano sulle competenze e sui processi aziendali; le prospettive relative alla piattaforma, alla sicurezza e alle operazioni si concentrano sulle competenze e sui processi tecnici. Ad esempio, la prospettiva relativa alle persone si rivolge alle parti interessate che gestiscono le risorse umane (HR), le funzioni del personale e la gestione del personale. In questa prospettiva, AWS CAF fornisce linee

guida per lo sviluppo delle persone, la formazione e le comunicazioni per aiutare a preparare l'organizzazione all'adozione del cloud di successo. Per ulteriori informazioni, consulta il sito web di AWS CAF e il white paper AWS CAF.

AWS Workload Qualification Framework (WQF)AWS

Uno strumento che valuta i carichi di lavoro di migrazione dei database, consiglia strategie di migrazione e fornisce stime del lavoro. AWS WQF è incluso in (). AWS Schema Conversion Tool AWS SCT Analizza gli schemi di database e gli oggetti di codice, il codice dell'applicazione, le dipendenze e le caratteristiche delle prestazioni e fornisce report di valutazione.

B

bot difettoso

Un bot che ha lo scopo di disturbare o causare danni a individui o organizzazioni.

BCP

Vedi la pianificazione della continuità operativa.

grafico comportamentale

Una vista unificata, interattiva dei comportamenti delle risorse e delle interazioni nel tempo. Puoi utilizzare un grafico comportamentale con Amazon Detective per esaminare tentativi di accesso non riusciti, chiamate API sospette e azioni simili. Per ulteriori informazioni, consulta Dati in un grafico comportamentale nella documentazione di Detective.

sistema big-endian

Un sistema che memorizza per primo il byte più importante. Vedi anche endianness.

Classificazione binaria

Un processo che prevede un risultato binario (una delle due classi possibili). Ad esempio, il modello di machine learning potrebbe dover prevedere problemi come "Questa e-mail è spam o non è spam?" o "Questo prodotto è un libro o un'auto?"

filtro Bloom

Una struttura di dati probabilistica ed efficiente in termini di memoria che viene utilizzata per verificare se un elemento fa parte di un set.

blue/green dispiegamento

Una strategia di implementazione in cui si creano due ambienti separati ma identici. La versione corrente dell'applicazione viene eseguita in un ambiente (blu) e la nuova versione dell'applicazione nell'altro ambiente (verde). Questa strategia consente di ripristinare rapidamente il sistema con un impatto minimo.

bot

Un'applicazione software che esegue attività automatizzate su Internet e simula l'attività o l'interazione umana. Alcuni bot sono utili o utili, come i web crawler che indicizzano le informazioni su Internet. Alcuni altri bot, noti come bot dannosi, hanno lo scopo di disturbare o causare danni a individui o organizzazioni.

botnet

Reti di bot infettate da malware e controllate da un'unica parte, nota come bot herder o bot operator. Le botnet sono il meccanismo più noto per scalare i bot e il loro impatto.

ramo

Un'area contenuta di un repository di codice. Il primo ramo creato in un repository è il ramo principale. È possibile creare un nuovo ramo a partire da un ramo esistente e quindi sviluppare funzionalità o correggere bug al suo interno. Un ramo creato per sviluppare una funzionalità viene comunemente detto ramo di funzionalità. Quando la funzionalità è pronta per il rilascio, il ramo di funzionalità viene ricongiunto al ramo principale. Per ulteriori informazioni, consulta Informazioni sulle filiali (documentazione). GitHub

accesso break-glass

In circostanze eccezionali e tramite una procedura approvata, un mezzo rapido per consentire a un utente di accedere a un sito a Account AWS cui in genere non dispone delle autorizzazioni necessarie. Per ulteriori informazioni, consulta l'indicatore Implementare le procedure break-glass nella guida. AWS Well-Architected

strategia brownfield

L'infrastruttura esistente nell'ambiente. Quando si adotta una strategia brownfield per un'architettura di sistema, si progetta l'architettura in base ai vincoli dei sistemi e dell'infrastruttura attuali. Per l'espansione dell'infrastruttura esistente, è possibile combinare strategie brownfield e greenfield.

cache del buffer

L'area di memoria in cui sono archiviati i dati a cui si accede con maggiore frequenza.

capacità di business

Azioni intraprese da un'azienda per generare valore (ad esempio vendite, assistenza clienti o marketing). Le architetture dei microservizi e le decisioni di sviluppo possono essere guidate dalle capacità aziendali. Per ulteriori informazioni, consulta la sezione Organizzazione in base alle funzionalità aziendali del whitepaper Esecuzione di microservizi containerizzati su AWS.

pianificazione della continuità operativa (BCP)

Un piano che affronta il potenziale impatto di un evento che comporta l'interruzione dell'attività, come una migrazione su larga scala, sulle operazioni e consente a un'azienda di riprendere rapidamente le operazioni.

C

CAF

Vedi AWS Cloud Adoption Framework.

implementazione canaria

Il rilascio lento e incrementale di una versione agli utenti finali. Quando sei sicuro, distribuisci la nuova versione e sostituisci la versione corrente nella sua interezza.

CoE

Vedi Cloud Center of Excellence.

CDC

Vedi Change Data Capture.

Change Data Capture (CDC)

Il processo di tracciamento delle modifiche a un'origine dati, ad esempio una tabella di database, e di registrazione dei metadati relativi alla modifica. È possibile utilizzare CDC per vari scopi, ad esempio il controllo o la replica delle modifiche in un sistema di destinazione per mantenere la sincronizzazione.

ingegneria del caos

Introduzione intenzionale di guasti o eventi dirompenti per testare la resilienza di un sistema. Puoi usare AWS Fault Injection Service (AWS FIS) per eseguire esperimenti che stressano i tuoi AWS carichi di lavoro e valutarne la risposta.

CI/CD

Vedi integrazione continua e distribuzione continua.

classificazione

Un processo di categorizzazione che aiuta a generare previsioni. I modelli di ML per problemi di classificazione prevedono un valore discreto. I valori discreti sono sempre distinti l'uno dall'altro. Ad esempio, un modello potrebbe dover valutare se in un'immagine è presente o meno un'auto.

Sviluppatore cittadino

Un utente aziendale che crea applicazioni di intelligenza artificiale utilizzando piattaforme senza code/low codice senza competenze tecniche specializzate.

crittografia lato client

Crittografia dei dati localmente, prima che il bersaglio li Servizio AWS riceva.

centro di eccellenza del cloud (CCoE)

Un team multidisciplinare che guida le iniziative di adozione del cloud in tutta l'organizzazione, tra cui lo sviluppo di best practice per il cloud, la mobilitazione delle risorse, la definizione delle tempistiche di migrazione e la guida dell'organizzazione attraverso trasformazioni su larga scala. Per ulteriori informazioni, consulta i post di CCoE sull' Cloud AWS Enterprise Strategy Blog.

cloud computing

La tecnologia cloud generalmente utilizzata per l'archiviazione remota di dati e la gestione dei dispositivi IoT. Il cloud computing è generalmente collegato alla tecnologia di edge computing.

modello operativo cloud

In un'organizzazione IT, il modello operativo utilizzato per creare, maturare e ottimizzare uno o più ambienti cloud. Per ulteriori informazioni, consulta Building your Cloud Operating Model.

fasi di adozione del cloud

Le quattro fasi che le organizzazioni in genere attraversano quando migrano verso Cloud AWS:

- Progetto: esecuzione di alcuni progetti relativi al cloud per scopi di dimostrazione e apprendimento
- Fondamento: effettuare investimenti fondamentali per dimensionare l'adozione del cloud (ad esempio, creazione di una zona di destinazione, definizione di un CCoE, definizione di un modello operativo)
- Migrazione: migrazione di singole applicazioni
- Re-invention — Ottimizzazione di prodotti e servizi e innovazione nel cloud

Queste fasi sono state definite da Stephen Orban nel post del blog The Journey Toward Cloud-First & the Stages of Adoption sul blog Enterprise Strategy. Cloud AWS Per informazioni sulla loro relazione con la strategia di AWS migrazione, consulta la guida alla preparazione alla migrazione.

CMDB

Vedi database di gestione della configurazione.

repository di codice

Una posizione in cui il codice di origine e altri asset, come documentazione, esempi e script, vengono archiviati e aggiornati attraverso processi di controllo delle versioni. Gli archivi cloud più comuni includono GitHub oBitbucket Cloud. Ogni versione del codice è denominata ramo. In una struttura a microservizi, ogni repository è dedicato a una singola funzionalità. Una singola CI/CD pipeline può utilizzare più repository.

cache fredda

Una cache del buffer vuota, non ben popolata o contenente dati obsoleti o irrilevanti. Ciò influisce sulle prestazioni perché l'istanza di database deve leggere dalla memoria o dal disco principale, il che richiede più tempo rispetto alla lettura dalla cache del buffer.

dati freddi

Dati a cui si accede raramente e che in genere sono storici. Quando si eseguono interrogazioni di questo tipo di dati, le interrogazioni lente sono in genere accettabili. Lo spostamento di questi dati su livelli o classi di storage meno costosi e con prestazioni inferiori può ridurre i costi.

visione artificiale (CV)

Un campo dell'intelligenza artificiale che utilizza l'apprendimento automatico per analizzare ed estrarre informazioni da formati visivi come immagini e video digitali. Ad esempio, Amazon SageMaker AI fornisce algoritmi di elaborazione delle immagini per CV.

deriva della configurazione

Per un carico di lavoro, una modifica della configurazione rispetto allo stato previsto. Potrebbe causare la non conformità del carico di lavoro e in genere è graduale e involontaria.

database di gestione della configurazione (CMDB)

Un repository che archivia e gestisce le informazioni su un database e il relativo ambiente IT, inclusi i componenti hardware e software e le relative configurazioni. In genere si utilizzano i dati di un CMDB nella fase di individuazione e analisi del portafoglio della migrazione.

Pacchetto di conformità

Una raccolta di AWS Config regole e azioni correttive che puoi assemblare per personalizzare i controlli di conformità e sicurezza. È possibile distribuire un pacchetto di conformità come singola entità in una regione Account AWS and o all'interno di un'organizzazione utilizzando un modello YAML. Per ulteriori informazioni, consulta i Conformance Pack nella documentazione. AWS Config

integrazione e distribuzione continue () CI/CD

Il processo di automazione delle fasi di origine, compilazione, test, gestione temporanea e produzione del processo di rilascio del software. CI/CD viene comunemente descritto come una pipeline. CI/CD può aiutarvi ad automatizzare i processi, migliorare la produttività, migliorare la qualità del codice e velocizzare le consegne. Per ulteriori informazioni, consulta Vantaggi della distribuzione continua. CD può anche significare continuous deployment (implementazione continua). Per ulteriori informazioni, consulta Distribuzione continua e implementazione continua a confronto.

CV

Vedi visione artificiale.

D

dati a riposo

Dati stazionari nella rete, ad esempio i dati archiviati.

classificazione dei dati

Un processo per identificare e classificare i dati nella rete in base alla loro criticità e sensibilità. È un componente fondamentale di qualsiasi strategia di gestione dei rischi di sicurezza informatica

perché consente di determinare i controlli di protezione e conservazione appropriati per i dati. La classificazione dei dati è un componente del pilastro della sicurezza nel AWS Well-Architected Framework. Per ulteriori informazioni, consulta Classificazione dei dati.

deriva dei dati

Una variazione significativa tra i dati di produzione e i dati utilizzati per addestrare un modello di machine learning o una modifica significativa dei dati di input nel tempo. La deriva dei dati può ridurre la qualità, l'accuratezza e l'equità complessive nelle previsioni dei modelli ML.

dati in transito

Dati che si spostano attivamente attraverso la rete, ad esempio tra le risorse di rete.

rete di dati

Un framework architettonico che fornisce la proprietà distribuita e decentralizzata dei dati con gestione e governance centralizzate.

riduzione al minimo dei dati

Il principio della raccolta e del trattamento dei soli dati strettamente necessari. Praticare la riduzione al minimo dei dati in the Cloud AWS può ridurre i rischi per la privacy, i costi e l'impronta di carbonio delle analisi.

perimetro dei dati

Una serie di barriere preventive nell' AWS ambiente che aiutano a garantire che solo le identità attendibili accedano alle risorse attendibili delle reti previste. Per ulteriori informazioni, consulta Building a data perimeter on. AWS

pre-elaborazione dei dati

Trasformare i dati grezzi in un formato che possa essere facilmente analizzato dal modello di ML. La pre-elaborazione dei dati può comportare la rimozione di determinate colonne o righe e l'eliminazione di valori mancanti, incoerenti o duplicati.

provenienza dei dati

Il processo di tracciamento dell'origine e della cronologia dei dati durante il loro ciclo di vita, ad esempio il modo in cui i dati sono stati generati, trasmessi e archiviati.

soggetto dei dati

Un individuo i cui dati vengono raccolti ed elaborati.

data warehouse

Un sistema di gestione dei dati che supporta la business intelligence, come l'analisi. I data warehouse contengono in genere grandi quantità di dati storici e vengono generalmente utilizzati per interrogazioni e analisi.

linguaggio di definizione del database (DDL)

Istruzioni o comandi per creare o modificare la struttura di tabelle e oggetti in un database.

linguaggio di manipolazione del database (DML)

Istruzioni o comandi per modificare (inserire, aggiornare ed eliminare) informazioni in un database.

DDL

Vedi linguaggio di definizione del database.

deep ensemble

Combinare più modelli di deep learning per la previsione. È possibile utilizzare i deep ensemble per ottenere una previsione più accurata o per stimare l'incertezza nelle previsioni.

deep learning

Un sottocampo del ML che utilizza più livelli di reti neurali artificiali per identificare la mappatura tra i dati di input e le variabili target di interesse.

difesa in profondità

Un approccio alla sicurezza delle informazioni in cui una serie di meccanismi e controlli di sicurezza sono accuratamente stratificati su una rete di computer per proteggere la riservatezza, l'integrità e la disponibilità della rete e dei dati al suo interno. Quando si adotta questa strategia AWS, si aggiungono più controlli a diversi livelli della AWS Organizations struttura per proteggere le risorse. Ad esempio, un approccio di difesa approfondita potrebbe combinare autenticazione a più fattori, segmentazione della rete e crittografia.

amministratore delegato

In AWS Organizations, un servizio compatibile può registrare un account AWS membro per amministrare gli account dell'organizzazione e gestire le autorizzazioni per quel servizio. Questo account è denominato amministratore delegato per quel servizio specifico. Per ulteriori informazioni e un elenco di servizi compatibili, consulta Servizi che funzionano con AWS Organizations nella documentazione di AWS Organizations .

implementazione

Il processo di creazione di un'applicazione, di nuove funzionalità o di correzioni di codice disponibili nell'ambiente di destinazione. L'implementazione prevede l'applicazione di modifiche in una base di codice, seguita dalla creazione e dall'esecuzione di tale base di codice negli ambienti applicativi.

Ambiente di sviluppo

Vedi ambiente.

controllo di rilevamento

Un controllo di sicurezza progettato per rilevare, registrare e avvisare dopo che si è verificato un evento. Questi controlli rappresentano una seconda linea di difesa e avvisano l'utente in caso di eventi di sicurezza che aggirano i controlli preventivi in vigore. Per ulteriori informazioni, consulta Controlli di rilevamento in Implementazione dei controlli di sicurezza in AWS.

mappatura del flusso di valore dello sviluppo (DVSM)

Un processo utilizzato per identificare e dare priorità ai vincoli che influiscono negativamente sulla velocità e sulla qualità nel ciclo di vita dello sviluppo del software. DVSM estende il processo di mappatura del flusso di valore originariamente progettato per pratiche di produzione snella. Si concentra sulle fasi e sui team necessari per creare e trasferire valore attraverso il processo di sviluppo del software.

gemello digitale

Una rappresentazione virtuale di un sistema reale, ad esempio un edificio, una fabbrica, un'attrezzatura industriale o una linea di produzione. I gemelli digitali supportano la manutenzione predittiva, il monitoraggio remoto e l'ottimizzazione della produzione.

tabella delle dimensioni

In uno schema a stella, una tabella più piccola che contiene gli attributi dei dati quantitativi in una tabella dei fatti. Gli attributi della tabella delle dimensioni sono in genere campi di testo o numeri discreti che si comportano come testo. Questi attributi vengono comunemente utilizzati per il vincolo delle query, il filtraggio e l'etichettatura dei set di risultati.

disastro

Un evento che impedisce a un carico di lavoro o a un sistema di raggiungere gli obiettivi aziendali nella sua sede principale di implementazione. Questi eventi possono essere disastri naturali,

guasti tecnici o il risultato di azioni umane, come errori di configurazione involontari o attacchi di malware.

disaster recovery (DR)

La strategia e il processo utilizzati per ridurre al minimo i tempi di inattività e la perdita di dati causati da un disastro. Per ulteriori informazioni, consulta Disaster Recovery of Workload su AWS: Recovery in the Cloud in the AWS Well-Architected Framework.

DML

Vedi linguaggio di manipolazione del database.

progettazione basata sul dominio

Un approccio allo sviluppo di un sistema software complesso collegandone i componenti a domini in evoluzione, o obiettivi aziendali principali, perseguiti da ciascun componente. Questo concetto è stato introdotto da Eric Evans nel suo libro Domain-Driven Design: Tackling Complexity in the Heart of Software (Boston: Addison-Wesley Professional, 2003). Per informazioni su come utilizzare la progettazione basata sul dominio con lo strangler fig pattern, consulta Modernizzare i servizi Web Microsoft ASP.NET (ASMX) legacy in modo incrementale utilizzando contenitori e Amazon API Gateway.

DOTT.

Vedi disaster recovery.

rilevamento della deriva

Tracciamento delle deviazioni da una configurazione di base. Ad esempio, puoi utilizzarlo AWS CloudFormation per rilevare la deriva nelle risorse di sistema oppure puoi usarlo AWS Control Tower per rilevare cambiamenti nella tua landing zone che potrebbero influire sulla conformità ai requisiti di governance.

DVSM

Vedi la mappatura del flusso di valore dello sviluppo.

E

EDA

Vedi analisi esplorativa dei dati.

MODIFICA

Vedi scambio elettronico di dati.

edge computing

La tecnologia che aumenta la potenza di calcolo per i dispositivi intelligenti all'edge di una rete IoT. Rispetto al cloud computing, l'edge computing può ridurre la latenza di comunicazione e migliorare i tempi di risposta.

scambio elettronico di dati (EDI)

Lo scambio automatizzato di documenti aziendali tra organizzazioni. Per ulteriori informazioni, vedere Cos'è lo scambio elettronico di dati.

crittografia

Un processo di elaborazione che trasforma i dati in chiaro, leggibili dall'uomo, in testo cifrato.

chiave crittografica

Una stringa crittografica di bit randomizzati generata da un algoritmo di crittografia. Le chiavi possono variare di lunghezza e ogni chiave è progettata per essere imprevedibile e univoca.

endianità

L'ordine in cui i byte vengono archiviati nella memoria del computer. Big-endian i sistemi memorizzano per primi il byte più importante. Little-endian i sistemi memorizzano per primi il byte meno importante.

endpoint

Vedi service endpoint.

servizio endpoint

Un servizio che puoi ospitare in un cloud privato virtuale (VPC) da condividere con altri utenti. Puoi creare un servizio endpoint con AWS PrivateLink e concedere autorizzazioni ad altri Account AWS o a AWS Identity and Access Management (IAM) principali. Questi account o principali possono connettersi al servizio endpoint in privato creando endpoint VPC di interfaccia. Per ulteriori informazioni, consulta Creazione di un servizio endpoint nella documentazione di Amazon Virtual Private Cloud (Amazon VPC).

pianificazione delle risorse aziendali (ERP)

Un sistema che automatizza e gestisce i processi aziendali chiave (come contabilità, MES e gestione dei progetti) per un'azienda.

crittografia envelope

Il processo di crittografia di una chiave di crittografia con un'altra chiave di crittografia. Per ulteriori informazioni, vedete Envelope encryption nella documentazione AWS Key Management Service (AWS KMS).

ambiente

Un'istanza di un'applicazione in esecuzione. Di seguito sono riportati i tipi di ambiente più comuni nel cloud computing:

- ambiente di sviluppo: un'istanza di un'applicazione in esecuzione disponibile solo per il team principale responsabile della manutenzione dell'applicazione. Gli ambienti di sviluppo vengono utilizzati per testare le modifiche prima di promuoverle negli ambienti superiori. Questo tipo di ambiente viene talvolta definito ambiente di test.
- ambienti inferiori: tutti gli ambienti di sviluppo di un'applicazione, ad esempio quelli utilizzati per le build e i test iniziali.
- ambiente di produzione: un'istanza di un'applicazione in esecuzione a cui gli utenti finali possono accedere. In una CI/CD pipeline, l'ambiente di produzione è l'ultimo ambiente di distribuzione.
- ambienti superiori: tutti gli ambienti a cui possono accedere utenti diversi dal team di sviluppo principale. Si può trattare di un ambiente di produzione, ambienti di preproduzione e ambienti per i test di accettazione da parte degli utenti.

epica

Nelle metodologie agili, categorie funzionali che aiutano a organizzare e dare priorità al lavoro. Le epiche forniscono una descrizione di alto livello dei requisiti e delle attività di implementazione. Ad esempio, le epopee della sicurezza AWS CAF includono la gestione delle identità e degli accessi, i controlli investigativi, la sicurezza dell'infrastruttura, la protezione dei dati e la risposta agli incidenti. Per ulteriori informazioni sulle epiche, consulta la strategia di migrazione AWS , consulta la guida all'implementazione del programma.

ERP

Vedi pianificazione delle risorse aziendali.

analisi esplorativa dei dati (EDA)

Il processo di analisi di un set di dati per comprenderne le caratteristiche principali. Si raccolgono o si aggregano dati e quindi si eseguono indagini iniziali per trovare modelli, rilevare anomalie

e verificare ipotesi. L'EDA viene eseguita calcolando statistiche di riepilogo e creando visualizzazioni di dati.

F

tabella dei fatti

Il tavolo centrale con schema a stella. Memorizza dati quantitativi sulle operazioni aziendali. In genere, una tabella dei fatti contiene due tipi di colonne: quelle che contengono misure e quelle che contengono una chiave esterna per una tabella di dimensioni.

fallire velocemente

Una filosofia che utilizza test frequenti e incrementali per ridurre il ciclo di vita dello sviluppo. È una parte fondamentale di un approccio agile.

limite di isolamento dei guasti

Nel Cloud AWS, un limite come una zona di disponibilità Regione AWS, un piano di controllo o un piano dati che limita l'effetto di un errore e aiuta a migliorare la resilienza dei carichi di lavoro. Per ulteriori informazioni, consulta AWS Fault Isolation Boundaries.

ramo di funzionalità

Vedi filiale.

caratteristiche

I dati di input che usi per fare una previsione. Ad esempio, in un contesto di produzione, le caratteristiche potrebbero essere immagini acquisite periodicamente dalla linea di produzione.

importanza delle caratteristiche

Quanto è importante una caratteristica per le previsioni di un modello. Di solito viene espresso come punteggio numerico che può essere calcolato con varie tecniche, come Shapley Additive Explanations (SHAP) e gradienti integrati. Per ulteriori informazioni, consulta Interpretabilità del modello di machine learning con AWS.

trasformazione delle funzionalità

Per ottimizzare i dati per il processo di machine learning, incluso l'arricchimento dei dati con fonti aggiuntive, il dimensionamento dei valori o l'estrazione di più set di informazioni da un singolo campo di dati. Ciò consente al modello di ML di trarre vantaggio dai dati. Ad esempio, se suddividi

la data "2021-05-27 00:15:37" in "2021", "maggio", "giovedì" e "15", puoi aiutare l'algoritmo di apprendimento ad apprendere modelli sfumati associati a diversi componenti dei dati.

prompt con pochi scatti

Fornire a un LLM un numero limitato di esempi che dimostrino l'attività e il risultato desiderato prima di chiedergli di eseguire un'attività simile. Questa tecnica è un'applicazione dell'apprendimento contestuale, in cui i modelli imparano da esempi (immagini) incorporati nei prompt. Few-shot i suggerimenti possono essere efficaci per attività che richiedono una formattazione, un ragionamento o una conoscenza del dominio specifici. Vedi anche zero-shot prompting.

FGAC

Vedi il controllo granulare degli accessi.

controllo granulare degli accessi (FGAC)

L'uso di più condizioni per consentire o rifiutare una richiesta di accesso.

migrazione flash-cut

Un metodo di migrazione del database che utilizza la replica continua dei dati tramite l'acquisizione dei dati delle modifiche per migrare i dati nel più breve tempo possibile, anziché utilizzare un approccio graduale. L'obiettivo è ridurre al minimo i tempi di inattività.

FM

Vedi il modello di base.

modello di fondazione (FM)

Una grande rete neurale di deep learning che si è addestrata su enormi set di dati generalizzati e non etichettati. Le FM sono in grado di eseguire un'ampia varietà di attività generali, come comprendere il linguaggio, generare testo e immagini e conversare in linguaggio naturale. Per ulteriori informazioni, consulta Cosa sono i modelli Foundation.

Gateway FM

Un intermediario centralizzato che controlla e normalizza l'accesso ai modelli di base. Conosciuto anche come gateway LLM.

G

IA generativa

Un sottoinsieme di modelli di intelligenza artificiale che sono stati addestrati su grandi quantità di dati e che possono utilizzare un semplice messaggio di testo per creare nuovi contenuti e artefatti, come immagini, video, testo e audio. Per ulteriori informazioni, consulta Cos'è l'IA generativa.

blocco geografico

Vedi restrizioni geografiche.

limitazioni geografiche (blocco geografico)

In Amazon CloudFront, un'opzione per impedire agli utenti di determinati paesi di accedere alle distribuzioni di contenuti. Puoi utilizzare un elenco consentito o un elenco di blocco per specificare i paesi approvati e vietati. Per ulteriori informazioni, consulta Limitare la distribuzione geografica dei contenuti nella CloudFront documentazione.

Flusso di lavoro di GitFlow

Un approccio in cui gli ambienti inferiori e superiori utilizzano rami diversi in un repository di codice di origine. Il flusso di lavoro Gitflow è considerato obsoleto e il flusso di lavoro basato su trunk è l'approccio moderno e preferito.

immagine dorata

Un'istantanea di un sistema o di un software che viene utilizzata come modello per distribuire nuove istanze di quel sistema o software. Ad esempio, nella produzione, un'immagine dorata può essere utilizzata per fornire software su più dispositivi e contribuire a migliorare la velocità, la scalabilità e la produttività nelle operazioni di produzione dei dispositivi.

strategia greenfield

L'assenza di infrastrutture esistenti in un nuovo ambiente. Quando si adotta una strategia greenfield per un'architettura di sistema, è possibile selezionare tutte le nuove tecnologie senza il vincolo della compatibilità con l'infrastruttura esistente, nota anche come brownfield. Per l'espansione dell'infrastruttura esistente, è possibile combinare strategie brownfield e greenfield.

guardrail

Una regola di livello elevato che consente di governare risorse, policy e conformità tra le unità organizzative (OU). I guardrail preventivi applicano le policy per garantire l'allineamento agli

standard di conformità. Vengono implementati utilizzando le policy di controllo dei servizi e i limiti delle autorizzazioni IAM. I guardrail di rilevamento rilevano le violazioni delle policy e i problemi di conformità e generano avvisi per porvi rimedio. Sono implementati utilizzando Amazon AWS Config AWS Security Hub CSPM GuardDuty AWS Trusted Advisor, Amazon Inspector e controlli personalizzati AWS Lambda .

guardrail (AI)

Meccanismi di sicurezza che filtrano, convalidano e limitano gli input e gli output degli agenti per contribuire a garantire un comportamento dell'IA responsabile e sicuro.

H

AH

Vedi disponibilità elevata.

migrazione di database eterogenea

Migrazione del database di origine in un database di destinazione che utilizza un motore di database diverso (ad esempio, da Oracle ad Amazon Aurora). La migrazione eterogenea fa in genere parte di uno sforzo di riprogettazione e la conversione dello schema può essere un'attività complessa. AWS offre AWS SCT che aiuta con le conversioni dello schema.

alta disponibilità (HA)

La capacità di un carico di lavoro di funzionare in modo continuo, senza intervento, in caso di sfide o disastri. I sistemi HA sono progettati per il failover automatico, fornire costantemente prestazioni di alta qualità e gestire carichi e guasti diversi con un impatto minimo sulle prestazioni.

modernizzazione storica

Un approccio utilizzato per modernizzare e aggiornare i sistemi di tecnologia operativa (OT) per soddisfare meglio le esigenze dell'industria manifatturiera. Uno storico è un tipo di database utilizzato per raccogliere e archiviare dati da varie fonti in una fabbrica.

dati di esclusione

Una parte di dati storici etichettati che viene trattenuta da un set di dati utilizzata per addestrare un modello di apprendimento automatico. È possibile utilizzare i dati di holdout per valutare le prestazioni del modello confrontando le previsioni del modello con i dati di holdout.

human-in-the-loop (HITL)

Un modello di flusso di lavoro in cui l'esecuzione degli agenti viene sospesa per la revisione e l'approvazione umana nei punti decisionali critici.

migrazione di database omogenea

Migrazione del database di origine in un database di destinazione che condivide lo stesso motore di database (ad esempio, da Microsoft SQL Server ad Amazon RDS per SQL Server). La migrazione omogenea fa in genere parte di un'operazione di rehosting o ridefinizione della piattaforma. Per migrare lo schema è possibile utilizzare le utilità native del database.

dati caldi

Dati a cui si accede frequentemente, ad esempio dati in tempo reale o dati di traduzione recenti. Questi dati richiedono in genere un livello o una classe di storage ad alte prestazioni per fornire risposte rapide alle query.

hotfix

Una soluzione urgente per un problema critico in un ambiente di produzione. A causa della sua urgenza, un hotfix viene in genere creato al di fuori del tipico DevOps flusso di lavoro di rilascio.

periodo di hypercare

Subito dopo la conversione, il periodo di tempo in cui un team di migrazione gestisce e monitora le applicazioni migrate nel cloud per risolvere eventuali problemi. In genere, questo periodo dura da 1 a 4 giorni. Al termine del periodo di hypercare, il team addetto alla migrazione in genere trasferisce la responsabilità delle applicazioni al team addetto alle operazioni cloud.

I

IaC

Vedi l'infrastruttura come codice.

Policy basata su identità

Una policy associata a uno o più principi IAM che definisce le relative autorizzazioni all'interno dell' Cloud AWS ambiente.

applicazione inattiva

Un'applicazione che prevede un uso di CPU e memoria medio compreso tra il 5% e il 20% in un periodo di 90 giorni. In un progetto di migrazione, è normale ritirare queste applicazioni o mantenerle on-premise.

IIoT

Vedi Industrial Internet of Things.

infrastruttura immutabile

Un modello che implementa una nuova infrastruttura per i carichi di lavoro di produzione anziché aggiornare, applicare patch o modificare l'infrastruttura esistente. Le infrastrutture immutabili sono intrinsecamente più coerenti, affidabili e prevedibili delle infrastrutture mutabili. Per ulteriori informazioni, consulta la best practice Deploy using immutable infrastructure nel Framework. AWS Well-Architected

VPC in ingresso (ingress)

In un'architettura AWS multi-account, un VPC che accetta, ispeziona e indirizza le connessioni di rete dall'esterno di un'applicazione. Nel documento Architettura di riferimento per la sicurezza di AWS si consiglia di configurare l'account di rete con VPC in entrata, in uscita e di ispezione per proteggere l'interfaccia bidirezionale tra l'applicazione e Internet in generale.

migrazione incrementale

Una strategia di conversione in cui si esegue la migrazione dell'applicazione in piccole parti anziché eseguire una conversione singola e completa. Ad esempio, inizialmente potresti spostare solo alcuni microservizi o utenti nel nuovo sistema. Dopo aver verificato che tutto funzioni correttamente, puoi spostare in modo incrementale microservizi o utenti aggiuntivi fino alla disattivazione del sistema legacy. Questa strategia riduce i rischi associati alle migrazioni di grandi dimensioni.

Industria 4.0

Un termine introdotto da Klaus Schwab nel 2016 per riferirsi alla modernizzazione dei processi di produzione attraverso progressi in termini di connettività, dati in tempo reale, automazione, analisi e. AI/ML

infrastruttura

Tutte le risorse e gli asset contenuti nell'ambiente di un'applicazione.

infrastruttura come codice (IaC)

Il processo di provisioning e gestione dell'infrastruttura di un'applicazione tramite un insieme di file di configurazione. Il processo IaC è progettato per aiutarti a centralizzare la gestione dell'infrastruttura, a standardizzare le risorse e a dimensionare rapidamente, in modo che i nuovi ambienti siano ripetibili, affidabili e coerenti.

Internet delle cose industriale (IIoT)

L'uso di sensori e dispositivi connessi a Internet nei settori industriali, come quello manifatturiero, energetico, automobilistico, sanitario, delle scienze della vita e dell'agricoltura. Per ulteriori informazioni, consulta Creazione di una strategia di trasformazione digitale dell'Internet delle cose industriale (IIoT).

VPC di ispezione

In un'architettura AWS multi-account, un VPC centralizzato che gestisce le ispezioni del traffico di rete tra VPC (uguali o diversi Regioni AWS), Internet e reti locali. Nel documento Architettura di riferimento per la sicurezza di AWS si consiglia di configurare l'account di rete con VPC in entrata, in uscita e di ispezione per proteggere l'interfaccia bidirezionale tra l'applicazione e Internet in generale.

Internet of Things (IoT)

La rete di oggetti fisici connessi con sensori o processori incorporati che comunicano con altri dispositivi e sistemi tramite Internet o una rete di comunicazione locale. Per ulteriori informazioni, consulta Cos'è l'IoT?

interpretabilità

Una caratteristica di un modello di machine learning che descrive il grado in cui un essere umano è in grado di comprendere in che modo le previsioni del modello dipendono dai suoi input. Per ulteriori informazioni, consulta Interpretabilità del modello di machine learning con. AWS

IoT

Vedi Internet of Things.

libreria di informazioni IT (ITIL)

Una serie di best practice per offrire servizi IT e allinearli ai requisiti aziendali. ITIL fornisce le basi per ITSM.

gestione dei servizi IT (ITSM)

Attività associate alla progettazione, implementazione, gestione e supporto dei servizi IT per un'organizzazione. Per informazioni sull'integrazione delle operazioni cloud con gli strumenti ITSM, consulta la guida all'integrazione delle operazioni.

ITIL

Vedi la libreria di informazioni IT.

ITSM

Vedi Gestione dei servizi IT.

L

controllo degli accessi basato su etichette (LBAC)

Un'implementazione del controllo di accesso obbligatorio (MAC) in cui agli utenti e ai dati stessi viene assegnato esplicitamente un valore di etichetta di sicurezza. L'intersezione tra l'etichetta di sicurezza utente e l'etichetta di sicurezza dei dati determina quali righe e colonne possono essere visualizzate dall'utente.

zona di destinazione

Una landing zone è un AWS ambiente multi-account ben progettato, scalabile e sicuro. Questo è un punto di partenza dal quale le organizzazioni possono avviare e distribuire rapidamente carichi di lavoro e applicazioni con fiducia nel loro ambiente di sicurezza e infrastruttura. Per ulteriori informazioni sulle zone di destinazione, consulta la sezione Configurazione di un ambiente AWS multi-account sicuro e scalabile.

modello linguistico di grandi dimensioni (LLM)

Un modello di intelligenza artificiale di deep learning preaddestrato su una grande quantità di dati. Un LLM può svolgere più attività, come rispondere a domande, riepilogare documenti, tradurre testo in altre lingue e completare frasi. Per ulteriori informazioni, consulta Cosa sono gli LLM.

migrazione su larga scala

Una migrazione di 300 o più server.

BIANCO

Vedi controllo degli accessi basato su etichette.

Privilegio minimo

La best practice di sicurezza per la concessione delle autorizzazioni minime richieste per eseguire un'attività. Per ulteriori informazioni, consulta Applicazione delle autorizzazioni del privilegio minimo nella documentazione di IAM.

eseguire il rehosting (lift and shift)

Vedi 7 R.

sistema little-endian

Un sistema che memorizza per primo il byte meno importante. Vedi anche endianità.

LLM

Vedi modello linguistico di grandi dimensioni.

ambienti inferiori

Vedi ambiente.

M

machine learning (ML)

Un tipo di intelligenza artificiale che utilizza algoritmi e tecniche per il riconoscimento e l'apprendimento di schemi. Il machine learning analizza e apprende dai dati registrati, come i dati dell'Internet delle cose (IoT), per generare un modello statistico basato su modelli. Per ulteriori informazioni, consulta la sezione Machine learning.

ramo principale

Vedi filiale.

malware

Software progettato per compromettere la sicurezza o la privacy del computer. Il malware potrebbe interrompere i sistemi informatici, divulgare informazioni sensibili o ottenere accessi non autorizzati. Esempi di malware includono virus, worm, ransomware, trojan horse, spyware e keylogger.

servizi gestiti

Servizi AWS per cui AWS gestisce il livello di infrastruttura, il sistema operativo e le piattaforme e si accede agli endpoint per archiviare e recuperare i dati. Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) e Amazon DynamoDB sono esempi di servizi gestiti. Questi sono noti anche come servizi astratti.

sistema di esecuzione della produzione (MES)

Un sistema software per tracciare, monitorare, documentare e controllare i processi di produzione che convertono le materie prime in prodotti finiti in officina.

MAP

Vedi Migration Acceleration Program.

MCP

Vedi Model Context Protocol.

Model Context Protocol (MCP)

Un protocollo stateless per la comunicazione tra agenti e strumenti.

Server MCP

Un servizio che espone uno o più strumenti tramite il Model Context Protocol.

meccanismo

Un processo completo in cui si crea uno strumento, si promuove l'adozione dello strumento e quindi si esaminano i risultati per apportare le modifiche. Un meccanismo è un ciclo che si rafforza e si migliora man mano che funziona. Per ulteriori informazioni, vedete Creazione di meccanismi nel AWS Well-Architected Framework.

account membro

Tutti gli account Account AWS diversi dall'account di gestione che fanno parte di un'organizzazione in AWS Organizations. Un account può essere membro di una sola organizzazione alla volta.

MEH

Vedi sistema di esecuzione della produzione.

Message Queuing Telemetry Transport (MQTT)

Un protocollo di comunicazione da macchina a macchina (M2M) leggero, basato sul publish/subscribemodello, per dispositivi IoT con risorse limitate.

microservizio

Un piccolo servizio indipendente che comunica tramite API ben definite ed è in genere di proprietà di piccoli team autonomi. Ad esempio, un sistema assicurativo potrebbe includere microservizi che si riferiscono a funzionalità aziendali, come vendite o marketing, o sottodomini, come acquisti, reclami o analisi. I vantaggi dei microservizi includono agilità, dimensionamento flessibile, facilità di implementazione, codice riutilizzabile e resilienza. Per ulteriori informazioni, consulta Integrazione dei microservizi utilizzando servizi serverless. AWS

architettura di microservizi

Un approccio alla creazione di un'applicazione con componenti indipendenti che eseguono ogni processo applicativo come microservizio. Questi microservizi comunicano tramite un'interfaccia ben definita utilizzando API leggere. Ogni microservizio in questa architettura può essere aggiornato, distribuito e dimensionato per soddisfare la richiesta di funzioni specifiche di un'applicazione. Per ulteriori informazioni, vedere Implementazione dei microservizi su. AWS

Programma di accelerazione della migrazione (MAP)

Un AWS programma che fornisce consulenza, supporto, formazione e servizi per aiutare le organizzazioni a costruire una solida base operativa per il passaggio al cloud e per contribuire a compensare il costo iniziale delle migrazioni. MAP include una metodologia di migrazione per eseguire le migrazioni precedenti in modo metodico e un set di strumenti per automatizzare e accelerare gli scenari di migrazione comuni.

migrazione su larga scala

Il processo di trasferimento della maggior parte del portfolio di applicazioni sul cloud avviene a ondate, con più applicazioni trasferite a una velocità maggiore in ogni ondata. Questa fase utilizza le migliori pratiche e le lezioni apprese nelle fasi precedenti per implementare una fabbrica di migrazione di team, strumenti e processi per semplificare la migrazione dei carichi di lavoro attraverso l'automazione e la distribuzione agile. Questa è la terza fase della strategia di migrazione AWS.

fabbrica di migrazione

Cross-functional team che semplificano la migrazione dei carichi di lavoro attraverso approcci automatizzati e agili. I team di Migration Factory includono in genere operazioni, analisti e

proprietari aziendali, ingegneri addetti alla migrazione, sviluppatori e DevOps professionisti che lavorano nell'ambito degli sprint. Tra il 20% e il 50% di un portfolio di applicazioni aziendali è costituito da schemi ripetuti che possono essere ottimizzati con un approccio di fabbrica. Per ulteriori informazioni, consulta la discussione sulle fabbriche di migrazione e la Guida alla fabbrica di migrazione al cloud in questo set di contenuti.

metadati di migrazione

Le informazioni sull'applicazione e sul server necessarie per completare la migrazione. Ogni modello di migrazione richiede un set diverso di metadati di migrazione. Esempi di metadati di migrazione includono la sottorete, il gruppo di sicurezza e l'account di destinazione. AWS

modello di migrazione

Un'attività di migrazione ripetibile che descrive in dettaglio la strategia di migrazione, la destinazione della migrazione e l'applicazione o il servizio di migrazione utilizzati. Esempio: riorganizza la migrazione su Amazon EC2 AWS con Application Migration Service.

Valutazione del portfolio di migrazione (MPA)

Uno strumento online che fornisce informazioni per la convalida del business case per la migrazione a. Cloud AWS MPA offre una valutazione dettagliata del portfolio (dimensionamento corretto dei server, prezzi, confronto del TCO, analisi dei costi di migrazione) e pianificazione della migrazione (analisi e raccolta dei dati delle applicazioni, raggruppamento delle applicazioni, prioritizzazione delle migrazioni e pianificazione delle ondate). Lo strumento MPA (richiede l'accesso) è disponibile gratuitamente per tutti i AWS consulenti e i consulenti dei partner APN.

valutazione della preparazione alla migrazione (MRA)

Il processo di acquisizione di informazioni sullo stato di preparazione al cloud di un'organizzazione, l'identificazione dei punti di forza e di debolezza e la creazione di un piano d'azione per colmare le lacune identificate, utilizzando il CAF. AWS Per ulteriori informazioni, consulta la guida di preparazione alla migrazione. MRA è la prima fase della strategia di migrazione AWS.

strategia di migrazione

L'approccio utilizzato per migrare un carico di lavoro verso. Cloud AWS Per ulteriori informazioni, consulta la voce 7 R in questo glossario e consulta Mobilita la tua organizzazione per accelerare le migrazioni su larga scala.

ML

Vedi machine learning.

modernizzazione

Trasformazione di un'applicazione obsoleta (legacy o monolitica) e della relativa infrastruttura in un sistema agile, elastico e altamente disponibile nel cloud per ridurre i costi, aumentare l'efficienza e sfruttare le innovazioni. Per ulteriori informazioni, vedere Strategia per la modernizzazione delle applicazioni in. Cloud AWS

valutazione della preparazione alla modernizzazione

Una valutazione che aiuta a determinare la preparazione alla modernizzazione delle applicazioni di un'organizzazione, identifica vantaggi, rischi e dipendenze e determina in che misura l'organizzazione può supportare lo stato futuro di tali applicazioni. Il risultato della valutazione è uno schema dell'architettura di destinazione, una tabella di marcia che descrive in dettaglio le fasi di sviluppo e le tappe fondamentali del processo di modernizzazione e un piano d'azione per colmare le lacune identificate. Per ulteriori informazioni, vedere Valutazione della preparazione alla modernizzazione per le applicazioni in. Cloud AWS

applicazioni monolitiche (monoliti)

Applicazioni eseguite come un unico servizio con processi strettamente collegati. Le applicazioni monolitiche presentano diversi inconvenienti. Se una funzionalità dell'applicazione registra un picco di domanda, l'intera architettura deve essere dimensionata. L'aggiunta o il miglioramento delle funzionalità di un'applicazione monolitica diventa inoltre più complessa man mano che la base di codice cresce. Per risolvere questi problemi, puoi utilizzare un'architettura di microservizi. Per ulteriori informazioni, consulta la sezione Scomposizione dei monoliti in microservizi.

MAPPA

Vedi Migration Portfolio Assessment.

MQTT

Vedi Message Queuing Telemetry Transport.

classificazione multiclasse

Un processo che aiuta a generare previsioni per più classi (prevedendo uno o più di due risultati). Ad esempio, un modello di machine learning potrebbe chiedere "Questo prodotto è un libro, un'auto o un telefono?" oppure "Quale categoria di prodotti è più interessante per questo cliente?"

infrastruttura mutabile

Un modello che aggiorna e modifica l'infrastruttura esistente per i carichi di lavoro di produzione. Per migliorare la coerenza, l'affidabilità e la prevedibilità, il AWS Well-Architected Framework consiglia l'uso di un'infrastruttura immutabile come best practice.

O

OAC

Vedi Origin Access Control.

QUERCIA

Vedi Origin Access Identity.

OCM

Vedi gestione delle modifiche organizzative.

migrazione offline

Un metodo di migrazione in cui il carico di lavoro di origine viene eliminato durante il processo di migrazione. Questo metodo prevede tempi di inattività prolungati e viene in genere utilizzato per carichi di lavoro piccoli e non critici.

OI

Vedi l'integrazione delle operazioni.

OLA

Vedi accordo a livello operativo.

migrazione online

Un metodo di migrazione in cui il carico di lavoro di origine viene copiato sul sistema di destinazione senza essere messo offline. Le applicazioni connesse al carico di lavoro possono continuare a funzionare durante la migrazione. Questo metodo comporta tempi di inattività pari a zero o comunque minimi e viene in genere utilizzato per carichi di lavoro di produzione critici.

OPC-UA

Vedi Open Process Communications - Unified Architecture.

Comunicazioni a processo aperto - Architettura unificata () OPC-UA

Un protocollo di comunicazione da macchina a macchina (M2M) per l'automazione industriale. OPC-UA fornisce uno standard di interoperabilità con schemi di crittografia, autenticazione e autorizzazione dei dati.

accordo a livello operativo (OLA)

Un accordo che chiarisce quali sono gli impegni reciproci tra i gruppi IT funzionali, a supporto di un accordo sul livello di servizio (SLA).

revisione della prontezza operativa (ORR)

Un elenco di domande e best practice associate che aiutano a comprendere, valutare, prevenire o ridurre la portata degli incidenti e dei possibili guasti. Per ulteriori informazioni, vedere Operational Readiness Reviews (ORR) nel Framework. AWS Well-Architected

tecnologia operativa (OT)

Sistemi hardware e software che interagiscono con l'ambiente fisico per controllare operazioni, apparecchiature e infrastrutture industriali. Nella produzione, l'integrazione di sistemi OT e di tecnologia dell'informazione (IT) è un obiettivo chiave per le trasformazioni dell'Industria 4.0.

integrazione delle operazioni (OI)

Il processo di modernizzazione delle operazioni nel cloud, che prevede la pianificazione, l'automazione e l'integrazione della disponibilità. Per ulteriori informazioni, consulta la guida all'integrazione delle operazioni.

trail organizzativo

Un percorso creato da noi AWS CloudTrail che registra tutti gli eventi di un'organizzazione per tutti Account AWS . AWS Organizations Questo percorso viene creato in ogni Account AWS che fa parte dell'organizzazione e tiene traccia dell'attività in ogni account. Per ulteriori informazioni, consulta Creazione di un percorso per un'organizzazione nella CloudTrail documentazione.

gestione del cambiamento organizzativo (OCM)

Un framework per la gestione di trasformazioni aziendali importanti e che comportano l'interruzione delle attività dal punto di vista delle persone, della cultura e della leadership. OCM aiuta le organizzazioni a prepararsi e passare a nuovi sistemi e strategie accelerando l'adozione del cambiamento, affrontando i problemi di transizione e promuovendo cambiamenti culturali e organizzativi. Nella strategia di AWS migrazione, questo framework si chiama accelerazione delle

persone, a causa della velocità di cambiamento richiesta nei progetti di adozione del cloud. Per ulteriori informazioni, consultare la Guida OCM.

controllo dell'accesso all'origine (OAC)

In CloudFront, un'opzione avanzata per limitare l'accesso per proteggere i contenuti di Amazon Simple Storage Service (Amazon S3). OAC supporta in tutto tutti i bucket S3 Regioni AWS, la crittografia lato server con AWS KMS (SSE-KMS) e le richieste dinamiche PUT e dirette al bucket S3. DELETE

identità di accesso origine (OAI)

Nel CloudFront, un'opzione per limitare l'accesso per proteggere i tuoi contenuti Amazon S3. Quando usi OAI, CloudFront crea un principale con cui Amazon S3 può autenticarsi. I principali autenticati possono accedere ai contenuti in un bucket S3 solo tramite una distribuzione specifica. CloudFront Vedi anche OAC, che fornisce un controllo degli accessi più granulare e avanzato.

ORR

Vedi la revisione della prontezza operativa.

- NON

Vedi la tecnologia operativa.

VPC in uscita (egress)

In un'architettura AWS multi-account, un VPC che gestisce le connessioni di rete avviate dall'interno di un'applicazione. Nel documento Architettura di riferimento per la sicurezza di AWS si consiglia di configurare l'account di rete con VPC in entrata, in uscita e di ispezione per proteggere l'interfaccia bidirezionale tra l'applicazione e Internet in generale.

P

limite delle autorizzazioni

Una policy di gestione IAM collegata ai principali IAM per impostare le autorizzazioni massime che l'utente o il ruolo possono avere. Per ulteriori informazioni, consulta Limiti delle autorizzazioni nella documentazione di IAM.

informazioni di identificazione personale (PII)

Informazioni che, se visualizzate direttamente o abbinate ad altri dati correlati, possono essere utilizzate per dedurre ragionevolmente l'identità di un individuo. Esempi di informazioni personali includono nomi, indirizzi e informazioni di contatto.

Informazioni che consentono l'identificazione personale degli utenti

Visualizza le informazioni di identificazione personale.

playbook

Una serie di passaggi predefiniti che raccolgono il lavoro associato alle migrazioni, come l'erogazione delle funzioni operative principali nel cloud. Un playbook può assumere la forma di script, runbook automatici o un riepilogo dei processi o dei passaggi necessari per gestire un ambiente modernizzato.

PLC

Vedi controllore logico programmabile.

PLM

Vedi la gestione del ciclo di vita del prodotto.

policy

Un oggetto in grado di definire le autorizzazioni (vedi politica basata sull'identità), specificare le condizioni di accesso (vedi politicabasata sulle risorse) o definire le autorizzazioni massime per tutti gli account di un'organizzazione in (vedi politica di controllo dei servizi). AWS Organizations

persistenza poliglotta

Scelta indipendente della tecnologia di archiviazione di dati di un microservizio in base ai modelli di accesso ai dati e ad altri requisiti. Se i microservizi utilizzano la stessa tecnologia di archiviazione di dati, possono incontrare problemi di implementazione o registrare prestazioni scadenti. I microservizi vengono implementati più facilmente e ottengono prestazioni e scalabilità migliori se utilizzano l'archivio dati più adatto alle loro esigenze.

valutazione del portfolio

Un processo di scoperta, analisi e definizione delle priorità del portfolio di applicazioni per pianificare la migrazione. Per ulteriori informazioni, consulta la pagina Valutazione della preparazione alla migrazione.

predicate

Una condizione di interrogazione che restituisce o, in genere, si trova in una clausola`WHERE`. `true` `false`

predicato pushdown

Una tecnica di ottimizzazione delle query del database che filtra i dati della query prima del trasferimento. Ciò riduce la quantità di dati che devono essere recuperati ed elaborati dal database relazionale e migliora le prestazioni delle query.

controllo preventivo

Un controllo di sicurezza progettato per impedire il verificarsi di un evento. Questi controlli sono la prima linea di difesa per impedire accessi non autorizzati o modifiche indesiderate alla rete. Per ulteriori informazioni, consulta Controlli preventivi in Implementazione dei controlli di sicurezza in AWS.

principale

Un'entità in AWS grado di eseguire azioni e accedere alle risorse. Questa entità è in genere un utente root per un Account AWS ruolo IAM o un utente. Per ulteriori informazioni, consulta Principali in Termini e concetti dei ruoli nella documentazione di IAM.

privacy fin dalla progettazione

Un approccio ingegneristico dei sistemi che tiene conto della privacy durante l'intero processo di sviluppo.

zone ospitate private

Un container che contiene informazioni su come si desidera che Amazon Route 53 risponda alle query DNS per un dominio e i relativi sottodomini all'interno di uno o più VPC. Per ulteriori informazioni, consulta Utilizzo delle zone ospitate private nella documentazione di Route 53.

controllo proattivo

Un controllo di sicurezza progettato per impedire l'implementazione di risorse non conformi. Questi controlli analizzano le risorse prima del loro provisioning. Se la risorsa non è conforme al controllo, non viene fornita. Per ulteriori informazioni, consulta la guida di riferimento sui controlli nella AWS Control Tower documentazione e consulta Controlli proattivi in Implementazione dei controlli di sicurezza su. AWS

gestione del ciclo di vita del prodotto (PLM)

La gestione dei dati e dei processi di un prodotto durante l'intero ciclo di vita, dalla progettazione, sviluppo e lancio, attraverso la crescita e la maturità, fino al declino e alla rimozione.

Ambiente di produzione

Vedi ambiente.

controllore logico programmabile (PLC)

Nella produzione, un computer altamente affidabile e adattabile che monitora le macchine e automatizza i processi di produzione.

concatenamento rapido

Utilizzo dell'output di un prompt LLM come input per il prompt successivo per generare risposte migliori. Questa tecnica viene utilizzata per suddividere un'attività complessa in sottoattività o per perfezionare o espandere iterativamente una risposta preliminare. Aiuta a migliorare l'accuratezza e la pertinenza delle risposte di un modello e consente risultati più granulari e personalizzati.

pseudonimizzazione

Il processo di sostituzione degli identificatori personali in un set di dati con valori segnaposto. La pseudonimizzazione può aiutare a proteggere la privacy personale. I dati pseudonimizzati sono ancora considerati dati personali.

publish/subscribe (pub/sub)

Un modello che consente comunicazioni asincrone tra microservizi per migliorare la scalabilità e la reattività. Ad esempio, in un MES basato su microservizi, un microservizio può pubblicare messaggi di eventi su un canale a cui altri microservizi possono abbonarsi. Il sistema può aggiungere nuovi microservizi senza modificare il servizio di pubblicazione.

Q

Piano di query

Una serie di passaggi, come le istruzioni, utilizzati per accedere ai dati in un sistema di database relazionale SQL.

regressione del piano di query

Quando un ottimizzatore del servizio di database sceglie un piano non ottimale rispetto a prima di una determinata modifica all'ambiente di database. Questo può essere causato da modifiche a statistiche, vincoli, impostazioni dell'ambiente, associazioni dei parametri di query e aggiornamenti al motore di database.

R

Matrice RACI

Vedi responsabile, responsabile, consultato, informato (RACI).

RAG

Vedi Retrieval Augmented Generation.

ransomware

Un software dannoso progettato per bloccare l'accesso a un sistema informatico o ai dati fino a quando non viene effettuato un pagamento.

Matrice RASCI

Vedi responsabile, responsabile, consultato, informato (RACI).

RCAC

Vedi controllo dell'accesso a righe e colonne.

replica di lettura

Una copia di un database utilizzata per scopi di sola lettura. È possibile indirizzare le query alla replica di lettura per ridurre il carico sul database principale.

riprogettare

Vedi 7 Rs.

obiettivo del punto di ripristino (RPO)

Il periodo di tempo massimo accettabile dall'ultimo punto di ripristino dei dati. Questo determina ciò che si considera una perdita di dati accettabile tra l'ultimo punto di ripristino e l'interruzione del servizio.

obiettivo del tempo di ripristino (RTO)

Il ritardo massimo accettabile tra l'interruzione del servizio e il ripristino del servizio.

rifattorizzare

Vedi 7 R.

Region

Una raccolta di AWS risorse in un'area geografica. Ciascuna Regione AWS è isolata e indipendente dalle altre per fornire tolleranza agli errori, stabilità e resilienza. Per ulteriori informazioni, consulta Specificare cosa può usare Regioni AWS il tuo account.

regressione

Una tecnica di ML che prevede un valore numerico. Ad esempio, per risolvere il problema "A che prezzo verrà venduta questa casa?" un modello di ML potrebbe utilizzare un modello di regressione lineare per prevedere il prezzo di vendita di una casa sulla base di dati noti sulla casa (ad esempio, la metratura).

riospitare

Vedi 7 R.

rilascio

In un processo di implementazione, l'atto di promuovere modifiche a un ambiente di produzione.

trasferisco

Vedi 7 Rs.

ripiattaforma

Vedi 7 Rs.

riacquisto

Vedi 7 Rs.

resilienza

La capacità di un'applicazione di resistere o ripristinare le interruzioni. L'elevata disponibilità e il disaster recovery sono considerazioni comuni quando si pianifica la resilienza in. Cloud AWSPer ulteriori informazioni, vedere Cloud AWS Resilience.

policy basata su risorse

Una policy associata a una risorsa, ad esempio un bucket Amazon S3, un endpoint o una chiave di crittografia. Questo tipo di policy specifica a quali principali è consentito l'accesso, le azioni supportate e qualsiasi altra condizione che deve essere soddisfatta.

matrice di assegnazione di responsabilità (RACI)

Una matrice che definisce i ruoli e le responsabilità di tutte le parti coinvolte nelle attività di migrazione e nelle operazioni cloud. Il nome della matrice deriva dai tipi di responsabilità definiti nella matrice: responsabile (R), responsabile (A), consultato (C) e informato (I). Il tipo di supporto (S) è facoltativo. Se includi il supporto, la matrice viene chiamata matrice RASCI e, se la escludi, viene chiamata matrice RACI.

controllo reattivo

Un controllo di sicurezza progettato per favorire la correzione di eventi avversi o deviazioni dalla baseline di sicurezza. Per ulteriori informazioni, consulta Controlli reattivi in Implementazione dei controlli di sicurezza in AWS.

retain

Vedi 7 R.

andare in pensione

Vedi 7 Rs.

Retrieval Augmented Generation (RAG)

Una tecnologia di intelligenza artificiale generativa in cui un LLM fa riferimento a una fonte di dati autorevole esterna alle sue fonti di dati di formazione prima di generare una risposta. Ad esempio, un modello RAG potrebbe eseguire una ricerca semantica nella knowledge base o nei dati personalizzati di un'organizzazione. Per ulteriori informazioni, consulta Cos'è il RAG.

rotazione

Processo di aggiornamento periodico di un segreto per rendere più difficile l'accesso alle credenziali da parte di un utente malintenzionato.

controllo dell'accesso a righe e colonne (RCAC)

L'uso di espressioni SQL di base e flessibili con regole di accesso definite. RCAC è costituito da autorizzazioni di riga e maschere di colonna.

RPO

Vedi obiettivo del punto di ripristino.

VERSO

Vedi obiettivo del tempo di ripristino.

runbook

Un insieme di procedure manuali o automatizzate necessarie per eseguire un'attività specifica. In genere sono progettati per semplificare operazioni o procedure ripetitive con tassi di errore elevati.

S

SAML 2.0

Uno standard aperto utilizzato da molti provider di identità (IdPs). Questa funzionalità abilita il single sign-on (SSO) federato, in modo che gli utenti possano accedere Console di gestione AWS o chiamare le operazioni AWS API senza che tu debba creare un utente in IAM per tutti i membri dell'organizzazione. Per ulteriori informazioni sulla federazione basata su SAML 2.0, consulta Informazioni sulla federazione basata su SAML 2.0 nella documentazione di IAM.

SCADA

Vedi controllo di supervisione e acquisizione dati.

SCP

Vedi la politica di controllo del servizio.

Secret

In AWS Secrets Manager, informazioni riservate o riservate, come una password o le credenziali utente, archiviate in forma crittografata. È costituito dal valore segreto e dai relativi metadati. Il valore segreto può essere binario, una stringa singola o più stringhe. Per ulteriori informazioni, consulta Cosa c'è in un segreto di Secrets Manager? nella documentazione di Secrets Manager.

sicurezza fin dalla progettazione

Un approccio di ingegneria dei sistemi che tiene conto della sicurezza durante l'intero processo di sviluppo.

controllo di sicurezza

Un guardrail tecnico o amministrativo che impedisce, rileva o riduce la capacità di un autore di minacce di sfruttare una vulnerabilità di sicurezza. Esistono quattro tipi principali di controlli di sicurezza: preventivi, investigativi, reattivi e proattivi.

rafforzamento della sicurezza

Il processo di riduzione della superficie di attacco per renderla più resistente agli attacchi. Può includere azioni come la rimozione di risorse che non sono più necessarie, l'implementazione di best practice di sicurezza che prevedono la concessione del privilegio minimo o la disattivazione di funzionalità non necessarie nei file di configurazione.

sistema di gestione delle informazioni e degli eventi di sicurezza (SIEM)

Strumenti e servizi che combinano sistemi di gestione delle informazioni di sicurezza (SIM) e sistemi di gestione degli eventi di sicurezza (SEM). Un sistema SIEM raccoglie, monitora e analizza i dati da server, reti, dispositivi e altre fonti per rilevare minacce e violazioni della sicurezza e generare avvisi.

automazione della risposta alla sicurezza

Un'azione predefinita e programmata progettata per rispondere o porre rimedio automaticamente a un evento di sicurezza. Queste automazioni fungono da controlli di sicurezza investigativi o reattivi che aiutano a implementare le migliori pratiche di sicurezza. AWS Esempi di azioni di risposta automatizzate includono la modifica di un gruppo di sicurezza VPC, l'applicazione di patch a un'istanza Amazon EC2 o la rotazione delle credenziali.

Crittografia lato server

Crittografia dei dati a destinazione, da parte di chi li riceve. Servizio AWS

Policy di controllo dei servizi (SCP)

Una policy che fornisce il controllo centralizzato sulle autorizzazioni per tutti gli account di un'organizzazione in AWS Organizations. Le SCP definiscono i guardrail o fissano i limiti alle azioni che un amministratore può delegare a utenti o ruoli. Puoi utilizzare le SCP come elenchi consentiti o elenchi di rifiuto, per specificare quali servizi o azioni sono consentiti o proibiti. Per ulteriori informazioni, consulta le politiche di controllo del servizio nella AWS Organizations documentazione.

endpoint del servizio

L'URL del punto di ingresso per un Servizio AWS. Puoi utilizzare l'endpoint per connetterti a livello di programmazione al servizio di destinazione. Per ulteriori informazioni, consulta Endpoint del Servizio AWS nei Riferimenti generali di AWS.

accordo sul livello di servizio (SLA)

Un accordo che chiarisce ciò che un team IT promette di offrire ai propri clienti, ad esempio l'operatività e le prestazioni del servizio.

indicatore del livello di servizio (SLI)

Misurazione di un aspetto prestazionale di un servizio, ad esempio il tasso di errore, la disponibilità o la velocità effettiva.

obiettivo a livello di servizio (SLO)

Una metrica target che rappresenta lo stato di un servizio, misurato da un indicatore del livello di servizio.

Modello di responsabilità condivisa

Un modello che descrive la responsabilità condivisa AWS per la sicurezza e la conformità del cloud. AWS è responsabile della sicurezza del cloud, mentre tu sei responsabile della sicurezza nel cloud. Per ulteriori informazioni, consulta Modello di responsabilità condivisa.

Shadow AI

Applicazioni di intelligenza artificiale non autorizzate create o utilizzate al di fuori dei canali regolamentati all'interno di un'organizzazione.

SIEM

Vedi il sistema di gestione delle informazioni e degli eventi sulla sicurezza.

punto di errore singolo (SPOF)

Un guasto in un singolo componente critico di un'applicazione che può disturbare il sistema.

SLAM

Vedi il contratto sul livello di servizio.

SLI

Vedi l'indicatore del livello di servizio.

LENTA

Vedi obiettivo del livello di servizio.

modello split-and-seed

Un modello per dimensionare e accelerare i progetti di modernizzazione. Man mano che vengono definite nuove funzionalità e versioni dei prodotti, il team principale si divide per creare nuovi team di prodotto. Questo aiuta a dimensionare le capacità e i servizi dell'organizzazione, migliora la produttività degli sviluppatori e supporta una rapida innovazione. Per ulteriori informazioni, vedere Approccio graduale alla modernizzazione delle applicazioni in. Cloud AWS

SPOF

Vedi punto di errore singolo.

schema a stella

Una struttura organizzativa di database che utilizza un'unica tabella dei fatti di grandi dimensioni per archiviare i dati transazionali o misurati e utilizza una o più tabelle dimensionali più piccole per memorizzare gli attributi dei dati. Questa struttura è progettata per l'uso in un data warehouse o per scopi di business intelligence.

modello del fico strangolatore

Un approccio alla modernizzazione dei sistemi monolitici mediante la riscrittura e la sostituzione incrementali delle funzionalità del sistema fino alla disattivazione del sistema legacy. Questo modello utilizza l'analogia di una pianta di fico che cresce fino a diventare un albero robusto e alla fine annienta e sostituisce il suo ospite. Il modello è stato introdotto da Martin Fowler come metodo per gestire il rischio durante la riscrittura di sistemi monolitici. Per un esempio di come applicare questo modello, consulta Modernizzare i servizi Web Microsoft ASP.NET (ASMX) legacy in modo incrementale utilizzando contenitori e Amazon API Gateway.

sottorete

Un intervallo di indirizzi IP nel VPC. Una sottorete deve risiedere in una singola zona di disponibilità.

controllo di supervisione e acquisizione dati (SCADA)

Nella produzione, un sistema che utilizza hardware e software per monitorare gli asset fisici e le operazioni di produzione.

crittografia simmetrica

Un algoritmo di crittografia che utilizza la stessa chiave per crittografare e decrittografare i dati.

test sintetici

Test di un sistema in modo da simulare le interazioni degli utenti per rilevare potenziali problemi o monitorare le prestazioni. Puoi usare Amazon CloudWatch Synthetics per creare questi test.

prompt di sistema

Una tecnica per fornire contesto, istruzioni o linee guida a un LLM per indirizzarne il comportamento. I prompt di sistema aiutano a impostare il contesto e stabilire regole per le interazioni con gli utenti.

T

tag

Key-value coppie che fungono da metadati per l'organizzazione delle risorse. AWS Con i tag è possibile a gestire, identificare, organizzare, cercare e filtrare le risorse. Per ulteriori informazioni, consulta Tagging delle risorse AWS.

variabile di destinazione

Il valore che stai cercando di prevedere nel machine learning supervisionato. Questo è indicato anche come variabile di risultato. Ad esempio, in un ambiente di produzione la variabile di destinazione potrebbe essere un difetto del prodotto.

elenco di attività

Uno strumento che viene utilizzato per tenere traccia dei progressi tramite un runbook. Un elenco di attività contiene una panoramica del runbook e un elenco di attività generali da completare. Per ogni attività generale, include la quantità stimata di tempo richiesta, il proprietario e lo stato di avanzamento.

ambiente di test

Vedi ambiente.

training

Fornire dati da cui trarre ispirazione dal modello di machine learning. I dati di training devono contenere la risposta corretta. L'algoritmo di apprendimento trova nei dati di addestramento i

pattern che mappano gli attributi dei dati di input al target (la risposta che si desidera prevedere). Produce un modello di ML che acquisisce questi modelli. Puoi quindi utilizzare il modello di ML per creare previsioni su nuovi dati di cui non si conosce il target.

strumento

Una funzione o API che un agente può richiamare per eseguire operazioni in sistemi esterni.

Transit Gateway

Un hub di transito di rete che è possibile utilizzare per collegare i VPC e le reti on-premise. Per ulteriori informazioni, consulta Cos'è un gateway di transito nella AWS Transit Gateway documentazione.

flusso di lavoro basato su trunk

Un approccio in cui gli sviluppatori creano e testano le funzionalità localmente in un ramo di funzionalità e quindi uniscono tali modifiche al ramo principale. Il ramo principale viene quindi integrato negli ambienti di sviluppo, preproduzione e produzione, in sequenza.

Accesso attendibile

Concessione delle autorizzazioni a un servizio specificato dall'utente per eseguire attività all'interno dell'organizzazione AWS Organizations e nei suoi account per conto dell'utente. Il servizio attendibile crea un ruolo collegato al servizio in ogni account, quando tale ruolo è necessario, per eseguire attività di gestione per conto dell'utente. Per ulteriori informazioni, consulta Utilizzo AWS Organizations con altri AWS servizi nella AWS Organizations documentazione.

regolazione

Modificare alcuni aspetti del processo di training per migliorare la precisione del modello di ML. Ad esempio, puoi addestrare il modello di ML generando un set di etichette, aggiungendo etichette e quindi ripetendo questi passaggi più volte con impostazioni diverse per ottimizzare il modello.

team da due pizze

Una piccola DevOps squadra che puoi sfamare con due pizze. Un team composto da due persone garantisce la migliore opportunità possibile di collaborazione nello sviluppo del software.

U

incertezza

Un concetto che si riferisce a informazioni imprecise, incomplete o sconosciute che possono minare l'affidabilità dei modelli di machine learning predittivi. Esistono due tipi di incertezza: l'incertezza epistemica, che è causata da dati limitati e incompleti, mentre l'incertezza aleatoria è causata dal rumore e dalla casualità insiti nei dati.

compiti indifferenziati

Conosciuto anche come sollevamento di carichi pesanti, è un lavoro necessario per creare e far funzionare un'applicazione, ma che non apporta valore diretto all'utente finale né offre vantaggi competitivi. Esempi di attività indifferenziate includono l'approvvigionamento, la manutenzione e la pianificazione della capacità.

ambienti superiori

Vedi ambiente.

V

vacuum

Un'operazione di manutenzione del database che prevede la pulizia dopo aggiornamenti incrementali per recuperare lo spazio di archiviazione e migliorare le prestazioni.

controllo delle versioni

Processi e strumenti che tengono traccia delle modifiche, ad esempio le modifiche al codice di origine in un repository.

Peering VPC

Una connessione tra due VPC che consente di instradare il traffico tramite indirizzi IP privati. Per ulteriori informazioni, consulta Che cos'è il peering VPC? nella documentazione di Amazon VPC.

vulnerabilità

Un difetto software o hardware che compromette la sicurezza del sistema.

W

cache calda

Una cache del buffer che contiene dati correnti e pertinenti a cui si accede frequentemente. L'istanza di database può leggere dalla cache del buffer, il che richiede meno tempo rispetto alla lettura dalla memoria dal disco principale.

dati caldi

Dati a cui si accede raramente. Quando si eseguono interrogazioni di questo tipo di dati, in genere sono accettabili interrogazioni moderatamente lente.

funzione finestra

Una funzione SQL che esegue un calcolo su un gruppo di righe che si riferiscono in qualche modo al record corrente. Le funzioni della finestra sono utili per l'elaborazione di attività, come il calcolo di una media mobile o l'accesso al valore delle righe in base alla posizione relativa della riga corrente.

Carico di lavoro

Una raccolta di risorse e codice che fornisce valore aziendale, ad esempio un'applicazione rivolta ai clienti o un processo back-end.

flusso di lavoro

Gruppi funzionali in un progetto di migrazione responsabili di una serie specifica di attività. Ogni flusso di lavoro è indipendente ma supporta gli altri flussi di lavoro del progetto. Ad esempio, il flusso di lavoro del portfolio è responsabile della definizione delle priorità delle applicazioni, della pianificazione delle ondate e della raccolta dei metadati di migrazione. Il flusso di lavoro del portfolio fornisce queste risorse al flusso di lavoro di migrazione, che quindi migra i server e le applicazioni.

VERME

Vedi scrivere una volta, leggere molti.

WQF

Vedi AWS Workload Qualification Framework.

scrivi una volta, leggi molte (WORM)

Un modello di storage che scrive i dati una sola volta e ne impedisce l'eliminazione o la modifica. Gli utenti autorizzati possono leggere i dati tutte le volte che è necessario, ma non possono modificarli. Questa infrastruttura di archiviazione dei dati è considerata immutabile.

Z

exploit zero-day

Un attacco, in genere malware, che sfrutta una vulnerabilità zero-day.

vulnerabilità zero-day

Un difetto o una vulnerabilità assoluta in un sistema di produzione. Gli autori delle minacce possono utilizzare questo tipo di vulnerabilità per attaccare il sistema. Gli sviluppatori vengono spesso a conoscenza della vulnerabilità causata dall'attacco.

prompt zero-shot

Fornire a un LLM le istruzioni per eseguire un'attività ma non esempi (immagini) che possano aiutarla. Il LLM deve utilizzare le sue conoscenze pre-addestrate per gestire l'attività. L'efficacia del prompt zero-shot dipende dalla complessità dell'attività e dalla qualità del prompt. Vedi anche few-shot prompting.

applicazione zombie

Un'applicazione che prevede un utilizzo CPU e memoria inferiore al 5%. In un progetto di migrazione, è normale ritirare queste applicazioni.

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