Opzioni avanzate di distribuzione delle applicazioni AMS

Guida per gli sviluppatori di applicazioni AMS Advanced

Version September 13, 2024

Guida per gli sviluppatori di applicazioni AMS Advanced: Opzioni avanzate di distribuzione delle applicazioni AMS

Table of Contents

Onboarding delle applicazioni

Benvenuto nel piano operativo AMS di AWS Managed Services (AMS). Lo scopo di questo documento è descrivere i vari metodi che è possibile utilizzare per l'onboarding delle applicazioni su AMS una volta configurata la gestione iniziale della rete e degli accessi, e i problemi da considerare nella scelta di tali metodi.

Questo documento è destinato agli integratori di sistemi e agli sviluppatori di applicazioni per fornire assistenza nella determinazione e nella creazione dei processi applicativi per i nuovi clienti AMS.

Che cos'è l'onboarding delle applicazioni?

L'onboarding delle applicazioni AMS si riferisce all'implementazione di risorse e applicazioni, secondo necessità, nell'infrastruttura AMS. Progettare applicazioni e infrastrutture sulla piattaforma AMS è molto simile a farlo su una piattaforma nativa. AWS Seguendo le migliori pratiche di progettazione di AWS applicazioni e infrastrutture e tenendo conto delle funzionalità fornite da AMS, si otterranno applicazioni capaci e utilizzabili ospitate nell'ambiente AMS.

> **Note**
>
> - Stati Uniti orientali (Virginia)
> - Stati Uniti occidentali (California settentrionale)
> - US West (Oregon)
> - Stati Uniti orientali (Ohio)
> - Canada (Centrale)
> - Sud America (San Paolo)
> - UE (Irlanda)
> - UE (Francoforte)
> - UE (Londra)
> - EU West (Parigi)
> - Asia Pacifico (Mumbai)
> - Asia Pacifico (Seoul)
> - Asia Pacifico (Singapore)

- Asia Pacifico (Sydney)
- Asia Pacifico (Tokyo)

Nuove regioni vengono aggiunte frequentemente. Per ulteriori informazioni, consulta Regioni AWS e zone di disponibilità.

Cosa facciamo, cosa non facciamo

AMS offre un approccio standardizzato all'implementazione dell'infrastruttura AWS e fornisce la necessaria gestione operativa continua. Per una descrizione completa dei ruoli, delle responsabilità e dei servizi supportati, consulta Descrizione del servizio.

> **Note**
>
> Per richiedere che AMS fornisca un servizio AWS aggiuntivo, invia una richiesta di servizio. Per ulteriori informazioni, consulta Making Service Requests.

- Cosa facciamo:

  Dopo aver completato l'onboarding, l'ambiente AMS è disponibile per ricevere richieste di modifica (RFCs), incidenti e richieste di assistenza. L'interazione con il servizio AMS ruota attorno al ciclo di vita di uno stack di applicazioni. I nuovi stack vengono ordinati da un elenco preconfigurato di modelli, lanciati in specifiche sottoreti di cloud privato virtuale (VPC), modificati durante la loro vita operativa tramite request for change (RFCs) e monitorati per eventi e incidenti 24 ore su 24, 7 giorni su 7.

  Gli stack di applicazioni attivi sono monitorati e mantenuti da AMS, comprese le patch, e non richiedono ulteriori azioni per tutta la durata dello stack, a meno che non sia necessaria una modifica o lo stack non venga smantellato. Gli incidenti rilevati da AMS che influiscono sullo stato e sul funzionamento dello stack generano una notifica e possono richiedere o meno l'intervento dell'utente per risolverli o verificarli. È possibile porre domande pratiche e altre richieste inviando una richiesta di assistenza.

  Inoltre, AMS consente di abilitare servizi AWS compatibili che non sono gestiti da AMS. Per informazioni sui servizi compatibili con AWS-AMS, consulta Modalità di provisioning self-service.

- Cosa NON facciamo:

  Sebbene AMS semplifica l'implementazione delle applicazioni fornendo una serie di opzioni manuali e automatizzate, sei responsabile dello sviluppo, del test, dell'aggiornamento e della gestione della tua applicazione. AMS fornisce assistenza per la risoluzione dei problemi di infrastruttura che influiscono sulle applicazioni, ma AMS non può accedere o convalidare le configurazioni delle applicazioni.

Immagini di macchine AMS Amazon (AMIs)

AMS produce Amazon Machine Images (AMIs) aggiornate ogni mese per i sistemi operativi supportati da AMS. Inoltre, AMS produce anche immagini di sicurezza avanzata (AMIs) basate sul benchmark CIS di livello 1 per un sottoinsieme di sistemi operativi supportati da AMS. Per scoprire quali sistemi operativi dispongono di un'immagine di sicurezza avanzata, consulta l'AMS Security User Guide, disponibile tramite la pagina AWS Artifact -> Reports (trova l'opzione Reports nel riquadro di navigazione a sinistra) filtrata per AWS Managed Services. Per accedere ad AWS Artifact, contatta il tuo CSDM per ricevere istruzioni o consulta la sezione Getting Started with AWS Artifact.

Per ricevere avvisi quando AMIs vengono rilasciati nuovi AMS, puoi abbonarti a un argomento di notifica di Amazon Simple Notification Service (Amazon SNS) chiamato «AMS AMI». Per maggiori dettagli, consulta Notifiche AMI AMS con SNS.

La convenzione di denominazione AMS AMI è:`customer-ams-<operating system>-<release date> - <version>`. (ad esempio,`customer-ams-rhel6-2018.11-3`)

Usa solo AMS AMIs che iniziano con`customer`.

AMS consiglia di utilizzare sempre l'AMI più recente. Puoi trovare la più recente in uno AMIs dei seguenti modi:

- Cercando nella console AMS, nella AMIspagina.
- Visualizzazione del file CSV AMI AMS più recente, disponibile dal CSDM o tramite questo file ZIP: contenuti AMI AMS 11.2024 e file CSV in un file ZIP.

  Per i file ZIP AMI precedenti, consulta la Cronologia dei documenti.
- Esecuzione di questo `SKMS` comando AMS (richiesto AMS SKMS SDK):

```
aws amsskms list-amis --vpc-id VPC_ID --query "Amis.sort_by(@,&Name)[?
 starts_with(Name,'customer')].[Name,AmiId,CreationTime]" --output table
```

Contenuto AMI AMS aggiunto alla base AWS AMIs, per sistema operativo (OS)

- Linux AMIs:
  - AWS Strumenti CLI
  - NTP
  - Agente del servizio Trend Micro Endpoint Protection
  - Implementazione del codice
  - PBIS/Beyond Trust AD Bridge
  - Agente SSM
  - Yum Upgrade per patch critiche
  - script e software di gestione AMS personalizzati (controllo dell'avvio, del join AD, del monitoraggio, della sicurezza e della registrazione)
- Server Windows: AMIs
  - Microsoft.NET Framework 4.5
  - PowerShell 5.1
  - AWS Strumenti per Windows PowerShell
  - PowerShell Moduli AMS che controllano l'avvio, il join AD, il monitoraggio, la sicurezza e la registrazione
  - Agente del servizio Trend Micro Endpoint Protection
  - Agente SSM
  - CloudWatch Agente
  - EC2Servizio Config (tramite Windows Server 2012 R2)
  - EC2Avvio (Windows Server 2016 e Windows Server 2019)
  - EC2LaunchV2 (Windows Server 2022 e versioni successive)

Basato AMIs su Linux:

- Amazon Linux 2023 (ultima versione secondaria) (AMI minima non supportata)

- Amazon Linux 2 (ultima versione secondaria)
- Amazon Linux (2ARM64)
- Red Hat Enterprise 7 (ultima versione secondaria)
- Red Hat Enterprise 8 (ultima versione secondaria)
- Red Hat Enterprise 9 (ultima versione secondaria)
- SUSE Linux Enterprise Server 15 SP6
- Ubuntu Linux 18.04
- Ubuntu Linux 20.04
- Ubuntu Linux 22.04
- Ubuntu Linux 24.04
- Amazon Linux: per una panoramica del prodotto, informazioni sui prezzi, informazioni sull'utilizzo e informazioni di supporto, consulta Amazon Linux AMI (HVM /64-bit) e Amazon Linux 2.

  Per ulteriori informazioni, consulta Amazon Linux 2 FAQs.
- RedHat Enterprise Linux (RHEL): per una panoramica del prodotto, informazioni sui prezzi, informazioni sull'utilizzo e informazioni di supporto, consulta Red Hat Enterprise Linux (RHEL) 7 (HVM).
- Ubuntu Linux 18.04: per una panoramica del prodotto, informazioni sui prezzi, informazioni sull'utilizzo e informazioni di supporto, consulta Ubuntu 18.04 LTS - Bionic.
- SUSE Linux Enterprise Server per applicazioni SAP 15: SP6
  - Esegui i seguenti passaggi una volta per account:
    1. Passare alla Marketplace AWS.
    2. Cerca il prodotto SUSE 15 SAP.
    3. Scegli Continua per abbonarti.
    4. Scegli Accetta i termini.
  - Completa i seguenti passaggi ogni volta che devi lanciare una nuova istanza di SUSE Linux Enterprise Server for SAP Applications 15 SP6:
    1. Nota l'ID AMI per l'AMI SUSE Linux Enterprise Server for SAP Applications 15 sottoscritta.
    2. Crea una distribuzione | Componenti stack avanzati | stack | Crea modifica di tipo EC2 ct-14027q0sjyt1h RFC. Sostituiscilo `InstanceAmiId` con l'ID Marketplace AWS AMI a cui ti sei abbonato.

Basato su Windows AMIs:

Microsoft Windows Server (2016, 2019 e 2022), basato sulla versione più recente di Windows AMIs.

Per esempi di creazione AMIs, consulta Creare AMI.

AMS per chi esce dall'imbarco: AMIs

AMS non comunica nulla AMIs all'utente durante l'offboarding per evitare ripercussioni sulle sue dipendenze. Se desideri rimuovere AMS AMIs dal tuo account, puoi utilizzare l'API per nascondere dati specifici. `cancel-image-launch-permission` AMIs Ad esempio, puoi utilizzare lo script seguente per nascondere tutti gli AMS AMIs che sono stati condivisi con il tuo account in precedenza:

```
for ami in $(aws ec2 describe-images --executable-users self --owners 027415890775 --
query 'Images[].ImageId' --output text) ;
    do
    aws ec2 cancel-image-launch-permission --image-id $ami ;
    done
```

È necessario che sia installata l'AWS CLI v2 affinché lo script possa essere eseguito senza errori. Per i passaggi di installazione dell'interfaccia a riga di comando di AWS, consulta Installazione o aggiornamento dell'ultima versione dell'interfaccia a riga di comando di AWS. Per i dettagli sul `cancel-image-launch-permission` comando, consulta. `cancel-image-launch-permission`

Sicurezza migliorata AMIs

AMS fornisce immagini di sicurezza avanzata (AMIs) basate sul benchmark CIS di livello 1 per un sottoinsieme dei sistemi operativi supportati da AMS. Per scoprire quali sistemi operativi dispongono di un'immagine di sicurezza avanzata, consulta la Guida alla sicurezza dei clienti di AWS Managed Services (AMS). Per accedere a questa guida, apri AWS Artifact, seleziona Report nel riquadro di navigazione a sinistra, quindi filtra per AWS Managed Services. Per istruzioni su come accedere AWS Artifact, contatta il tuo CSDM o consulta la sezione Getting Started with AWS Artifact per ulteriori informazioni.

Termini chiave AMS

- AMS Advanced: i servizi descritti nella sezione «Descrizione del servizio» della documentazione di AMS Advanced. Vedi Descrizione del servizio.

- Account AMS Advanced: AWS account che soddisfano sempre tutti i requisiti degli AMS Advanced Onboarding Requirements. Per informazioni sui vantaggi di AMS Advanced, sui case study e per contattare un addetto alle vendite, consulta AWS Managed Services.
- Account AMS Accelerate: AWS account che soddisfano in ogni momento tutti i requisiti degli AMS Accelerate Onboarding Requirements. Vedi Guida introduttiva ad AMS Accelerate.
- AWS Managed Services: AMS e/o AMS Accelerate.
- Account AWS Managed Services: gli account AMS e/o gli account AMS Accelerate.
- Raccomandazione critica: una raccomandazione emessa AWS tramite una richiesta di servizio che informa l'utente che è necessario intervenire per proteggersi da potenziali rischi o interruzioni delle risorse o del. Servizi AWS Se decidi di non seguire una raccomandazione critica entro la data specificata, sei l'unico responsabile di eventuali danni derivanti dalla tua decisione.
- Configurazione richiesta dal cliente: qualsiasi software, servizio o altra configurazione non identificata in:
  - Accelerazione: configurazioni supportate o AMS Accelerate; descrizione del servizio.
  - AMS Advanced: configurazioni supportate o AMS Advanced; descrizione del servizio.
- Comunicazione di un incidente: AMS comunica un Incidente all'utente o l'utente richiede un Incidente ad AMS tramite un Incidente creato nel Support Center for AMS Accelerate e nella console AMS per AMS. La console AMS Accelerate fornisce un riepilogo degli incidenti e delle richieste di assistenza sulla dashboard e collegamenti al Support Center per i dettagli.
- Ambiente gestito: gli account AMS Advanced e/o gli account AMS Accelerate gestiti da AMS.

  Per AMS Advanced, questi includono account multi-account landing zone (MALZ) e account single-account landing zone (SALZ).
- Data di inizio della fatturazione: il giorno lavorativo successivo alla AWS ricezione delle informazioni richieste nell'e-mail di onboarding di AWS Managed Services. L'e-mail di onboarding di AWS Managed Services si riferisce AWS all'e-mail inviata dall'utente per raccogliere le informazioni necessarie per attivare AWS Managed Services sui propri account.

  Per gli account registrati successivamente da te, la data di inizio della fatturazione è il giorno successivo all'invio da parte di AWS Managed Services di una notifica di attivazione di AWS Managed Services per l'account registrato. Una notifica di attivazione di AWS Managed Services si verifica quando:

  1. Concedi l'accesso a un AWS account compatibile e lo consegni a AWS Managed Services.
  2. AWS Managed Services progetta e crea l'account AWS Managed Services.

- Interruzione del servizio: puoi chiudere AWS Managed Services per tutti gli account AWS Managed Services o per uno specifico account AWS Managed Services per qualsiasi motivo fornendo un preavviso di AWS almeno 30 giorni tramite una richiesta di servizio. Alla data di cessazione del servizio, puoi:
  1. AWS ti cede i controlli di tutti gli account AWS Managed Services o degli account AWS Managed Services specificati, a seconda dei casi, oppure
  2. Le parti rimuovono i AWS Identity and Access Management ruoli che consentono AWS l'accesso da tutti gli account AWS Managed Services o dagli account AWS Managed Services specificati, a seconda dei casi.
- Data di cessazione del servizio: la data di cessazione del servizio è l'ultimo giorno del mese di calendario successivo alla fine del periodo di preavviso di cessazione richiesto di 30 giorni. Se la fine del periodo di disdetta richiesto cade dopo il ventesimo giorno del mese solare, la data di cessazione del servizio è l'ultimo giorno del mese di calendario successivo. Di seguito sono riportati alcuni esempi di scenari relativi alle date di cessazione.
  - Se l'avviso di risoluzione viene fornito il 12 aprile, il preavviso di 30 giorni termina il 12 maggio. La data di cessazione del servizio è il 31 maggio.
  - Se viene fornito un avviso di risoluzione il 29 aprile, il preavviso di 30 giorni termina il 29 maggio. La data di cessazione del servizio è il 30 giugno.
- Fornitura di AWS Managed Services: ti AWS mette a disposizione e puoi accedere e utilizzare AWS Managed Services per ogni account AWS Managed Services a partire dalla data di inizio del servizio.
- Chiusura per determinati account AWS Managed Services: puoi chiudere AWS Managed Services per uno specifico account AWS Managed Services per qualsiasi motivo fornendo un AWS preavviso tramite una richiesta di servizio («Richiesta di terminazione dell'account AMS»).

Termini di gestione degli incidenti:

- Evento: un cambiamento nell'ambiente AMS.
- Avviso: ogni volta che un evento di un Servizio AWS operatore supportato supera una soglia e attiva un allarme, viene creato un avviso e viene inviato un avviso all'elenco dei contatti. Inoltre, viene creato un incidente nell'elenco degli incidenti.
- Incidente: un'interruzione o un peggioramento non pianificato delle prestazioni dell'ambiente AMS o di AWS Managed Services che provoca un impatto come riportato da AWS Managed Services o da te.

- Problema: causa principale condivisa di uno o più incidenti.
- Risoluzione di un incidente o risoluzione di un incidente:
  - AMS ha ripristinato tutti i servizi o le risorse AMS non disponibili relativi all'incidente riportandoli allo stato disponibile, oppure
  - AMS ha stabilito che gli stack o le risorse non disponibili non possono essere ripristinati allo stato disponibile, oppure
  - AMS ha avviato un ripristino dell'infrastruttura autorizzato dall'utente.
- Tempo di risposta agli incidenti: la differenza di tempo tra il momento in cui si crea un incidente e il momento in cui AMS fornisce una risposta iniziale tramite console, e-mail, centro di assistenza o telefono.
- Tempo di risoluzione dell'incidente: la differenza di tempo tra il momento in cui AMS o l'utente creano un incidente e il momento in cui l'incidente viene risolto.
- Priorità dell'incidente: in che modo AMS o l'utente assegna la priorità agli incidenti, definendoli come bassa, media o alta.
  - Bassa: un problema non critico del servizio AMS.
  - Medio: un servizio AWS all'interno dell'ambiente gestito è disponibile ma non funziona come previsto (secondo la descrizione del servizio applicabile).
  - Alto: (1) la console AMS o uno o più AMS APIs nell'ambiente gestito non sono disponibili; oppure (2) uno o più stack o risorse AMS all'interno dell'ambiente gestito non sono disponibili e l'indisponibilità impedisce all'applicazione di svolgere la propria funzione.

  AMS può riclassificare gli incidenti in base alle linee guida di cui sopra.
- Ripristino dell'infrastruttura: ridistribuzione degli stack esistenti, sulla base di modelli degli stack interessati, e avvio di un ripristino dei dati basato sull'ultimo punto di ripristino noto, se non diversamente specificato dall'utente, quando la risoluzione degli incidenti non è possibile.

Termini relativi all'infrastruttura:

- Ambiente di produzione gestito: un account cliente in cui risiedono le applicazioni di produzione del cliente.
- Ambiente non di produzione gestito: un account cliente che contiene solo applicazioni non di produzione, come applicazioni per lo sviluppo e il test.
- Stack AMS: un gruppo di una o più AWS risorse gestite da AMS come singola unità.

- Infrastruttura immutabile: un modello di manutenzione dell'infrastruttura tipico EC2 dei gruppi Amazon Auto Scaling ASGs () in cui i componenti dell'infrastruttura aggiornati (ad AWS esempio l'AMI) vengono sostituiti per ogni implementazione, anziché essere aggiornati sul posto. Il vantaggio dell'infrastruttura immutabile è che tutti i componenti rimangono in uno stato sincrono poiché vengono sempre generati dalla stessa base. L'immutabilità è indipendente da qualsiasi strumento o flusso di lavoro per la creazione dell'AMI.
- Infrastruttura mutabile: un modello di manutenzione dell'infrastruttura tipico per gli stack che non sono gruppi di Amazon EC2 Auto Scaling e contengono una singola istanza o solo poche istanze. Questo modello rappresenta più da vicino l'implementazione di sistema tradizionale, basata su hardware, in cui un sistema viene distribuito all'inizio del suo ciclo di vita e poi gli aggiornamenti vengono distribuiti su quel sistema nel tempo. Tutti gli aggiornamenti del sistema vengono applicati alle istanze singolarmente e possono comportare tempi di inattività del sistema (a seconda della configurazione dello stack) a causa del riavvio dell'applicazione o del sistema.
- Gruppi di sicurezza: firewall virtuali per l'istanza per controllare il traffico in entrata e in uscita. I gruppi di sicurezza operano a livello di istanza, non di sottorete. Pertanto, a ciascuna istanza di una sottorete del VPC potrebbe essere assegnato un set diverso di gruppi di sicurezza.
- Accordi sul livello di servizio (SLAs): parte dei contratti AMS con voi che definiscono il livello di servizio previsto.
- SLA non disponibile e indisponibilità:
  - Una richiesta API da te inviata che genera un errore.
  - Una richiesta di console inviata da te che genera una risposta HTTP 5xx (il server non è in grado di eseguire la richiesta).
  - Tutte Servizio AWS le offerte che costituiscono stack o risorse nell'infrastruttura gestita da AMS si trovano in uno stato di «interruzione del servizio», come indicato nella Service Health Dashboard.
  - L'indisponibilità derivante direttamente o indirettamente da un'esclusione di AMS non viene considerata nel determinare l'idoneità ai crediti di servizio. I servizi sono considerati disponibili a meno che non soddisfino i criteri per l'indisponibilità.
- Obiettivi del livello di servizio (SLOs): parte dei contratti AMS con l'utente che definiscono obiettivi di servizio specifici per i servizi AMS.

Termini di applicazione delle patch:

- Patch obbligatorie: aggiornamenti di sicurezza critici per risolvere problemi che potrebbero compromettere lo stato di sicurezza dell'ambiente o dell'account. Un «aggiornamento critico di

sicurezza» è un aggiornamento di sicurezza classificato come «Critico» dal fornitore di un sistema operativo supportato da AMS.

- Patch annunciate o rilasciate: le patch vengono generalmente annunciate e rilasciate in base a una pianificazione. Le patch emergenti vengono annunciate quando viene scoperta la necessità della patch e, di solito, subito dopo, la patch viene rilasciata.
- Patch add-on: applicazione di patch basata su tag per le istanze AMS che sfrutta la funzionalità AWS Systems Manager (SSM) in modo da poter etichettare le istanze e applicare le patch a tali istanze utilizzando una linea di base e una finestra configurate dall'utente.
- Metodi di patch:
  - Patch sul posto: applicazione di patch che viene eseguita modificando le istanze esistenti.
  - Patch sostitutiva dell'AMI: patch che viene eseguita modificando il parametro di riferimento AMI di una configurazione di avvio del gruppo Auto EC2 Scaling esistente.
- Fornitore di patch (fornitori di sistemi operativi, terze parti): le patch vengono fornite dal fornitore o dall'organo direttivo dell'applicazione.
- Tipi di patch:
  - Aggiornamento critico di sicurezza (CSU): aggiornamento di sicurezza classificato come «critico» dal fornitore di un sistema operativo supportato.
  - Aggiornamento importante (UI): un aggiornamento di sicurezza classificato come «Importante» o un aggiornamento non relativo alla sicurezza classificato come «Critico» dal fornitore di un sistema operativo supportato.
  - Altro aggiornamento (OU): un aggiornamento del fornitore di un sistema operativo supportato che non è una CSU o un'interfaccia utente.
- Patch supportate: AMS supporta le patch a livello di sistema operativo. Gli aggiornamenti vengono rilasciati dal fornitore per correggere vulnerabilità di sicurezza o altri bug o per migliorare le prestazioni. Per un elenco delle configurazioni attualmente supportate OSs, consulta Support Configurations.

Termini di sicurezza:

- Detective Controls: una libreria di monitor creati o abilitati da AMS che forniscono una supervisione continua degli ambienti e dei carichi di lavoro gestiti dai clienti per configurazioni che non sono in linea con i controlli di sicurezza, operativi o dei clienti e intervengono informando i proprietari, modificando in modo proattivo o interrompendo le risorse.

Termini della richiesta di assistenza:

- Richiesta di assistenza: una richiesta da parte tua relativa a un'azione che desideri che AMS intraprenda per tuo conto.
- Notifica di avviso: avviso pubblicato da AMS nella pagina di elenco delle richieste di assistenza quando viene attivato un avviso AMS. Il contatto configurato per il tuo account viene inoltre avvisato tramite il metodo configurato (ad esempio, e-mail). Se hai dei tag di contatto sulle tue istanze/risorse e hai fornito il consenso al tuo cloud service delivery manager (CSDM) per le notifiche basate su tag, le informazioni di contatto (valore chiave) contenute nel tag vengono notificate anche per gli avvisi AMS automatici.
- Notifica di servizio: un avviso di AMS che viene pubblicato nella pagina dell'elenco delle richieste di assistenza.

Termini vari:

- Interfaccia AWS Managed Services: per AMS: la console AWS Managed Services Advanced, l'API AMS CM e Supporto l'API. Per AMS Accelerate: la Supporto console e Supporto l'API.
- Soddisfazione del cliente (CSAT): AMS CSAT viene informata con analisi approfondite, tra cui le valutazioni della corrispondenza dei casi su ogni caso o corrispondenza, sondaggi trimestrali e così via.
- DevOps: DevOps è una metodologia di sviluppo che promuove fortemente l'automazione e il monitoraggio in tutte le fasi. DevOps mira a cicli di sviluppo più brevi, a una maggiore frequenza di implementazione e a rilasci più affidabili, riunendo le funzioni di sviluppo e operazioni, tradizionalmente separate, su una base di automazione. Quando gli sviluppatori possono gestire le operazioni e le operazioni informano lo sviluppo, problemi e problemi vengono scoperti e risolti più rapidamente e gli obiettivi aziendali vengono raggiunti più rapidamente.
- ITIL: Information Technology Infrastructure Library (denominata ITIL) è un framework ITSM progettato per standardizzare il ciclo di vita dei servizi IT. ITIL è organizzato in cinque fasi che coprono il ciclo di vita dei servizi IT: strategia del servizio, progettazione del servizio, transizione del servizio, funzionamento del servizio e miglioramento del servizio.
- Gestione dei servizi IT (ITSM): un insieme di pratiche che allineano i servizi IT alle esigenze dell'azienda.
- Managed Monitoring Services (MMS): AMS gestisce il proprio sistema di monitoraggio, Managed Monitoring Service (MMS), che utilizza gli eventi AWS Health e aggrega i dati di CloudWatch Amazon e i dati di Servizi AWS altri, notificando agli operatori AMS (online 24 ore su 24, 7 giorni su

7) qualsiasi allarme creato tramite un argomento di Amazon Simple Notification Service (Amazon SNS).

- Namespace: quando crei policy IAM o lavori con Amazon Resource Names (ARNs), identifichi un utente Servizio AWS utilizzando uno spazio dei nomi. È possibile utilizzare gli spazi dei nomi quando si identificano azioni e risorse.

Qual è il mio modello operativo?

In qualità di cliente AMS, la tua organizzazione ha deciso di separare le operazioni relative alle applicazioni e all'infrastruttura e di utilizzare AMS per le operazioni infrastrutturali. AMS collaborerà con il team di progettazione e sviluppo delle applicazioni e con il team di progettazione dell'infrastruttura per garantire che le operazioni dell'infrastruttura funzionino senza intoppi. L'immagine seguente illustra questo concetto:

AMS si assume la responsabilità delle operazioni AWS dell'infrastruttura mentre i team sono responsabili delle operazioni applicative. In qualità di team di progettazione delle applicazioni e dell'infrastruttura, dovete capire chi gestirà l'applicazione una volta che sarà stata implementata in produzione nell'infrastruttura AMS. Questa guida illustra gli approcci comuni alla progettazione dell'infrastruttura in relazione alla distribuzione e alla manutenzione delle applicazioni.

Gestione dei servizi in AWS Managed Services

Argomenti

Come funziona il servizio AMS per te.

Governance degli account in AWS Managed Services

Questa sezione riguarda la governance degli account AMS.

Sei designato come cloud service delivery manager (CSDM) che fornisce assistenza consultiva in AMS e ha una conoscenza dettagliata del tuo caso d'uso e dell'architettura tecnologica per l'ambiente gestito. CSDMs collabora con account manager, account manager tecnici, architetti cloud AWS Managed Services (CAs) e architetti di soluzioni AWS (SAs), a seconda dei casi, per contribuire al lancio di nuovi progetti e fornire consigli sulle migliori pratiche durante lo sviluppo del software e i processi operativi. Il CSDM è il punto di contatto principale per AMS. Le principali responsabilità del vostro CSDM sono:

- Organizza e conduci riunioni mensili di revisione del servizio con i clienti.
- Fornisci dettagli sulla sicurezza, sugli aggiornamenti software per l'ambiente e sulle opportunità di ottimizzazione.
- Rispetta i tuoi requisiti, comprese le richieste di funzionalità per AMS.
- Rispondi e risolvi le richieste di fatturazione e segnalazione dei servizi.
- Fornisci informazioni dettagliate per consigli finanziari e di ottimizzazione della capacità.

Inizio del servizio in AWS Managed Services

Inizio del servizio: la data di inizio del servizio per un account AWS Managed Services è il primo giorno del primo mese di calendario dopo il quale AWS ti notifica che le attività stabilite nei requisiti di onboarding per quell'account AWS Managed Services sono state completate; a condizione che se AWS invia tale notifica dopo il ventesimo giorno di un mese solare, la data di inizio del servizio è il primo giorno del secondo mese di calendario successivo alla data di tale notifica.

Inizio del servizio

- R sta per parte responsabile che fa il lavoro per raggiungere l'obiettivo.
- I sta per informato; una parte che viene informata sui progressi, spesso solo al completamento del compito o del risultato.

Inizio del servizio

Fase #	Titolo della fase	Descrizione	Custome	AMBITI
1.	Consegna dell'account AWS al cliente	Il cliente crea un nuovo account AWS e lo consegna ad AWS Managed Services	R	I
2.	Account AWS Managed Services - progettazione	Completa la progettazione dell'account AWS Managed Services	I	R
3.	Account AWS Managed Services: crea	Un account AWS Managed Services viene creato in base alla progettazione nella fase 2.	I	R

Gestione delle relazioni con i clienti (CRM)

AWS Managed Services (AMS) fornisce un processo di gestione delle relazioni con i clienti (CRM) per garantire che venga stabilita e mantenuta una relazione ben definita con te. La base di questa relazione si basa sulla conoscenza approfondita di AMS delle tue esigenze aziendali. Il processo CRM facilita la comprensione accurata e completa di:

- Le esigenze della tua azienda e come soddisfarle
- Le tue capacità e i tuoi vincoli
- AMS e le tue diverse responsabilità e obblighi

Il processo CRM consente ad AMS di utilizzare metodi coerenti per fornirti servizi e garantire la governance del tuo rapporto con AMS. Il processo CRM include:

- Identificazione dei principali stakeholder
- Istituire un team di governance
- Conduzione e documentazione delle riunioni di revisione del servizio con voi
- Fornitura di una procedura formale per i reclami relativi al servizio con una procedura di intensificazione
- Implementazione e monitoraggio del processo di soddisfazione e feedback
- Gestione del contratto

Processo CRM

Il processo CRM include le seguenti attività:

- Identificazione e comprensione dei processi e delle esigenze aziendali. Il vostro accordo con AMS identifica i vostri stakeholder.
- Definizione dei servizi da fornire per soddisfare le vostre esigenze e requisiti.
- Vi incontreremo durante le riunioni di revisione del servizio per discutere di eventuali modifiche all'ambito del servizio AMS, allo SLA, al contratto e alle vostre esigenze aziendali. È possibile tenere riunioni intermedie con voi per discutere di prestazioni, risultati, problemi e piani d'azione.
- Monitoraggio della vostra soddisfazione utilizzando il nostro sondaggio sulla soddisfazione dei clienti e il feedback fornito durante le riunioni.
- Segnalazione delle prestazioni su report mensili sulle prestazioni misurati internamente.
- Esaminate insieme a voi il servizio per determinare le opportunità di miglioramento. Ciò include comunicazioni frequenti con l'utente in merito al livello e alla qualità del servizio AMS fornito.

Riunioni CRM

I responsabili della fornitura di servizi cloud AMS (CSDMs) organizzano riunioni con voi regolarmente per discutere dei percorsi di servizio (operazioni, sicurezza e innovazioni dei prodotti) e dei percorsi esecutivi (rapporti SLA, misure di soddisfazione e cambiamenti nelle esigenze aziendali).

Riunione	Scopo	Modalità	Partecipanti
Revisione settimanale dello stato (opzionale)	Problemi o incidenti in sospeso, patch, eventi di sicurezza, record dei problemi Tendenza operativa a 12 settimane (+/- 6) Preoccupazioni degli operatori dell'applicazione Programma del fine settimana	Cliente in loco location/ Telecom/Chime	AMS: CSDM e architetto cloud (CA) Membri del team assegnati dal cliente (ad es.: Cloud/Infrastructure, Application Support, team di architettura, ecc.)
Revisione aziendale mensile	Esamina le prestazioni dei livelli di servizio (report, analisi e tendenze) Analisi finanziaria Roadmap del prodotto GATTO	Cliente in loco location/ Telecom/Chime	AMS: CSDM, cloud architect (CA), account team AMS, responsabile tecnico del prodotto AMS (TPM) (opzionale), AMS OPS manager (opzionale) Tu: rappresentante dell'Application Operator

Riunione	Scopo	Modalità	Partecipanti
Revisione aziendale trimestrale	Prestazioni e tendenze di Scorecard e Service Level Agreement (SLA) (6 mesi) Prossimi piani/migrazioni per 3/6/9/12 mesi Rischio e mitigazione del rischio Principali iniziative di miglioramento Elementi della roadmap del prodotto Opportunità orientate al futuro Finanziari Iniziative di riduzione dei costi Ottimizzazione aziendale	Ubicazione del cliente in sede	AMS: CSDM, architetto cloud, account team AMS, direttore del servizio AMS, responsabile operativo AMS Tu: rappresentante dell'operatore dell'applicazione, rappresentante del servizio, direttore del servizio

Organizzazione delle riunioni CRM

L'AMS CSDM è responsabile della documentazione della riunione, tra cui:

- Creazione dell'agenda, che include le azioni da intraprendere, i problemi e l'elenco dei partecipanti.
- Creazione dell'elenco delle azioni esaminate in ogni riunione per garantire che le questioni vengano completate e risolte nei tempi previsti.
- Distribuzione dei verbali delle riunioni e dell'elenco delle azioni da intraprendere ai partecipanti alla riunione tramite e-mail entro un giorno lavorativo dalla riunione.
- Archiviazione dei verbali delle riunioni nell'apposito archivio di documenti.

In assenza del CSDM, il rappresentante AMS che dirige la riunione crea e distribuisce i verbali.

> **Note**
>
> Il CSDM collabora con voi per stabilire la governance degli account.

Rapporti mensili CRM

Il CSDM AMS prepara e invia presentazioni mensili sulle prestazioni del servizio. Le presentazioni includono informazioni su quanto segue:

- Data del rapporto
- Riepilogo e approfondimenti:
  - Richieste principali: numero di stack totali e attivi, stato dell'applicazione delle patch allo stack, stato di attivazione dell'account (solo durante l'onboarding), riepiloghi dei problemi specifici del cliente
  - Prestazioni: statistiche sulla risoluzione degli incidenti, avvisi, patch, richieste di modifica (), richieste di servizio e disponibilità di console e API RFCs
  - Problemi, sfide, preoccupazioni e rischi: stato dei problemi specifici del cliente
  - Prossimi articoli: piani di onboarding o risoluzione degli incidenti specifici per il cliente
- Risorse gestite: grafici e diagrammi a torta degli stack
- Metriche AMS: metriche di monitoraggio ed eventi, metriche degli incidenti, metriche di aderenza agli SLA di AMS, metriche delle richieste di assistenza, metriche di gestione delle modifiche, metriche di storage, metriche di continuità, metriche Trusted Advisor e riepiloghi dei costi (presentati in diversi modi). Richieste di funzionalità. Informazioni di contatto

> **Note**
>
> Oltre alle informazioni descritte, il CSDM vi informa anche di qualsiasi modifica sostanziale dell'ambito o dei termini, incluso il ricorso a subappaltatori da parte di AMS per attività operative.
> AMS genera report sull'applicazione di patch e backup che il CSDM include nel rapporto mensile. Come parte del sistema di generazione dei report, AMS aggiunge al tuo account un'infrastruttura che non è accessibile a te:
>
> - Un S3 Bucket, con i dati grezzi riportati
> - Un'istanza Athena, con definizioni di query per interrogare i dati

- Un Glue Crawler per leggere i dati grezzi dal bucket S3

Ottimizzazione dei costi in AWS Managed Services

AWS Managed Services fornisce ogni mese report dettagliati sull'utilizzo dei costi e sui risparmi durante le revisioni aziendali mensili (MBRs).

AMS segue una serie standard di processi e meccanismi per identificare le possibilità di riduzione dei costi nei tuoi account gestiti e aiutarti a pianificare e implementare le modifiche per ottimizzare la spesa in AWS.

> **Note**
>
> AMS sta sviluppando un video per contribuire all'ottimizzazione dei costi. Il primo passo consiste nel fornirti un PDF e un foglio di calcolo Excel con le migliori pratiche di ottimizzazione dei costi. Per accedere a queste risorse, aprite il file ZIP della guida rapida all'ottimizzazione dei costi.

Framework di ottimizzazione dei costi

AMS segue un approccio in tre fasi per ottimizzare i costi di AWS:

1. Identifica le modalità di ottimizzazione dei costi nel tuo ambiente gestito
2. Presentate un piano di ottimizzazione dei costi
3. Contribuisci a raggiungere l'ottimizzazione dei costi in modo misurabile

Identifica le vie di ottimizzazione dei costi nell'ambiente gestito

AMS utilizza strumenti AWS nativi come Cost explorer e Trusted Advisor, sfruttando al contempo oltre 20 modelli di risparmio sui costi tra ottimizzazione dell'architettura, ottimizzazioni delle EC2 istanze e ottimizzazioni AWS incentrate sull'account per creare consigli di risparmio su misura per te.

Alcuni dei consigli di ottimizzazione includono quanto segue.

Consigli per l'ottimizzazione dell'architettura:

- Utilizzo ottimale della classe di storage S3: Amazon S3 offre una gamma di classi di storage per soddisfare diversi requisiti di carico di lavoro in base all'accesso ai dati, alla resilienza e ai costi. L'analisi delle classi di storage S3 Intelligent-Tiering e S3 basata sulle esigenze del carico di lavoro consente di gestire i costi di S3 in modo efficiente.
- Utilizzo di architetture di caching: l'utilizzo delle istanze di cache, ove applicabile, può aiutarti a sostituire alcune istanze di database, soddisfacendo contemporaneamente i requisiti IOPS.
- Risparmi sugli upgrade EBS: la migrazione dei volumi EBS da gp2 a gp3 offre un risparmio sui costi fino al 20% e puoi sfruttare le prestazioni di base prevedibili di 3.000 IOPS e 125 MiB/s, indipendentemente dalle dimensioni del volume.
- Utilizzo dell'elasticità: le funzionalità di auto-scaling offerte AWS consentono un utilizzo efficace delle risorse e consentono l'ottimizzazione dei costi. La revisione e l'aggiornamento regolari delle politiche di scalabilità delle istanze in base alle necessità consentono ulteriori risparmi sui costi.

EC2 consigli incentrati sulle istanze

- Ridimensionamento corretto delle istanze: raccomandazioni incentrate sul dimensionamento delle istanze e sulle configurazioni ottimali in base all'utilizzo. I consigli includono anche l'utilizzo della funzionalità Amazon EC2 Auto Scaling e la EC2 sostituzione delle istanze, ove applicabile, AWS Lambda con contenuti Web statici su Amazon S3, ecc.
- Pianificazione delle istanze: l'utilizzo di AMS Resource Scheduler per avviare e arrestare automaticamente le istanze in base a una pianificazione temporale aiuta a contenere i costi, in particolare per le istanze non di produzione che non vengono utilizzate durante le ore non lavorative.
- Abbonamento ai piani di risparmio: il piano di risparmio è il modo più semplice per risparmiare sull'utilizzo. AWS Gli EC2 Instance Savings Plans offrono risparmi fino al 72% rispetto ai prezzi On-Demand sull'utilizzo delle EC2 istanze Amazon. Gli Amazon SageMaker AI Savings Plans offrono fino al 64% di risparmio sull'utilizzo dei servizi Amazon SageMaker AI. AMS fornisce raccomandazioni appropriate sui piani di risparmio in base all'utilizzo AWS delle risorse.
- Guida all'utilizzo e al consumo delle istanze EC2 riservate (RI): Amazon Reserved Instances (RI) offre uno sconto significativo (fino al 75%) rispetto ai prezzi On-Demand e fornisce una riserva di capacità se utilizzate in una zona di disponibilità specifica.
- Utilizzo delle istanze Spot: i carichi di lavoro con tolleranza ai guasti possono utilizzare le istanze Spot e ridurre i prezzi fino al 90%.
- Chiusura delle istanze inattive: identificazione e segnalazione delle istanze inattive o con un utilizzo limitato che possono essere interrotte.

Consigli incentrati sull'account

- Pulizia dell'account: a livello di account, AMS identifica anche i volumi EBS non utilizzati, i CloudTrail percorsi duplicati, gli account vuoti con risorse inutilizzate e così via, e fornisce consigli per la pulizia.
- Raccomandazioni sugli SLA: inoltre, AMS esamina regolarmente gli account Plus e Premium e consiglia di scegliere il livello SLA giusto per gli account.
- Ottimizzazione dell'automazione AMS: AMS ottimizza continuamente l'automazione AMS e l'infrastruttura utilizzata per fornire i servizi AMS.

Presentazione ai clienti e assistenza nella pianificazione

AMS effettua revisioni aziendali mensili (MBRs) con le principali parti interessate dei clienti e presenta le possibilità, i meccanismi e le raccomandazioni di riduzione dei costi identificati, oltre ai potenziali risparmi sui costi. Collaboriamo inoltre con voi per pianificare le modifiche necessarie.

Assisti nell'implementazione delle raccomandazioni e misura l'impatto sui costi

AMS aiuta a raggiungere e misurare gli impatti sui costi e le modifiche all'ottimizzazione.

Valuti l'impatto applicativo, il rischio e i criteri di successo delle modifiche consigliate e invii le richieste di modifica appropriate (RFCs) tramite la console AMS. AMS collabora con te e implementa le modifiche relative all'ottimizzazione dei costi nei tuoi account gestiti. AMS misura l'impatto sui costi e include i risparmi realizzati nelle revisioni aziendali mensili (). MBRs

Matrice di responsabilità per l'ottimizzazione dei costi

Responsabilità nell'ottimizzazione dei costi di AMS.

Ottimizzazione dei costi RACI

Attività	Customer	AMS
Compilazione di raccomandazioni per la riduzione	I	R

Attività	Customer	AMS
dei costi e preparazione del rapporto		
Presentazione del rapporto sui risparmi	C	R
Pianificazione delle modifiche associate ai risparmi sui costi	R	C
Valutazione dell'impatto e del rischio del cambiamento	R	C
Raccolta fondi RFCs per l'attuazione delle modifiche	R	C

Attività	Customer	AMS
Revisione RFCs e implementazione delle modifiche	C	R
Test dell'applicazione e convalida dell'implementazione delle modifiche	R	C
Misurazione dell'impatto sui costi dopo la modifica e presentazione al cliente	I	R

Orari di servizio in AWS Managed Services

Funzionalità	AMS Advanced
	Livello Premium
Richiesta di assistenza	24/7
Gestione degli incidenti (P2-P3)	24/7
Backup e ripristino	24/7
Gestione delle patch	24/7
Monitoraggio e avvisi	24/7
Richiesta di modifica automatica (RFC)	24/7
Richiesta di modifica non automatizzata (RFC)	24/7
Responsabile della fornitura di servizi cloud (CSDM)	Dal lunedì al venerdì: dalle 08:00 alle 17:00, orario lavorativo locale

Ottenere assistenza in AWS Managed Services

AMS ti supporta con la gestione degli incidenti, la gestione delle richieste di servizio e la gestione delle modifiche 24 ore al giorno, 7 giorni alla settimana, 365 giorni all'anno (in conformità con l'accordo sul livello di servizio AMS applicato all'account).

Per segnalare un problema di prestazioni del servizio AWS o AMS che ha un impatto sull'ambiente gestito, utilizza la console AMS e invia un rapporto sull'incidente. Per i dettagli, consulta Segnalazione di un incidente. Per informazioni generali sulla gestione degli incidenti con AMS, consulta Risposta agli incidenti.

Per richiedere informazioni o consigli o per richiedere servizi aggiuntivi ad AMS, utilizza la console AMS e invia una richiesta di servizio. Per i dettagli, vedi Creazione di una richiesta di assistenza. Per informazioni generali sulle richieste di assistenza AMS, vedere Gestione delle richieste di assistenza.

Sviluppo di applicazioni

Processi e pratiche di sviluppo delle applicazioni che consentono la progettazione e la distribuzione efficaci delle applicazioni in un ambiente AWS Managed Services (AMS). AMS ti guida attraverso il seguente processo di alto livello:

1. Immagina e progetta un'applicazione da sviluppare o integrare nel tuo ambiente gestito da AMS. Alcune considerazioni:
    a. Come installerai la tua applicazione? Con l'automazione utilizzando uno strumento di distribuzione come Ansible o manualmente caricando direttamente i file necessari?
    b. Come aggiornerai la tua applicazione? Con un approccio mutabile che aggiorna ogni istanza separatamente o con un approccio immutabile che aggiorna ogni istanza con un'unica AMI aggiornata in un gruppo di Auto Scaling?
2. Pianifica e progetta l'infrastruttura che verrà utilizzata per ospitare l'applicazione utilizzando librerie di AWS architettura, linee guida AWS «Well-Architected» e AMS e altri esperti in materia di architettura cloud. Le seguenti sezioni di questa guida forniscono informazioni utili in tal senso.
3. Seleziona un approccio di implementazione dell'infrastruttura:
    a. Full Stack: tutti i componenti dell'infrastruttura vengono distribuiti contemporaneamente, insieme.
    b. Tier and Tie: le implementazioni dell'infrastruttura vengono implementate separatamente e, successivamente, collegate alle modifiche dei gruppi di sicurezza. Questo tipo di implementazione si ottiene anche mediante una configurazione seriale dei componenti dello stack che si basa l'uno sull'altro; ad esempio, specificando il sistema di bilanciamento del carico creato in precedenza quando si crea un gruppo di Auto Scaling.
    c. Quali ambienti, come Dev, Staging e Prod, utilizzerai?
4. Scegliete i tipi di modifica AMS (CTs) che forniranno gli stack o i livelli necessari e prepareranno le richieste di modifica necessarie (). RFCs
5. Invia il RFCs comando per attivare l'implementazione dell'infrastruttura nell'ambiente appropriato.
6. Distribuisci l'applicazione utilizzando l'approccio di distribuzione dell'applicazione selezionato.
7. Rielabora l'infrastruttura e le applicazioni secondo necessità.
8. Implementa l'infrastruttura e le applicazioni in ambienti successivi appropriati, supponendo che la prima implementazione sia in un ambiente non di produzione.

9. La manutenzione continua è gestita da AMS che gestisce l'infrastruttura sottostante e dai team operativi che gestiscono le infrastrutture delle applicazioni.
10. Per disattivare un'applicazione, interrompi l'infrastruttura AMS corrispondente.

Essere ben architettati

AWS Riteniamo che sistemi ben architettati aumentino notevolmente le probabilità di successo aziendale. L'AWS Architecture Center fornisce una guida esperta sull'architettura in. Cloud AWS

Consigliamo i seguenti articoli e white paper per aiutarti a comprendere i pro e i contro delle decisioni che devi prendere durante la creazione di sistemi. AWS

Sei Well-Architected? : Presenta il AWS Well-Architected Framework, basato su sei pilastri:

- Eccellenza operativa: il pilastro dell'eccellenza operativa si concentra sulla gestione e sul monitoraggio dei sistemi per offrire valore aziendale e migliorare continuamente processi e procedure. Gli argomenti chiave includono la gestione e l'automazione delle modifiche, la risposta agli eventi e la definizione di standard per gestire con successo le operazioni quotidiane.
- Sicurezza: il pilastro della sicurezza si concentra sulla protezione di informazioni e sistemi. Gli argomenti chiave includono la riservatezza e l'integrità dei dati, l'identificazione e la gestione di chi può fare cosa con la gestione delle autorizzazioni, la protezione dei sistemi e l'istituzione di controlli per rilevare gli eventi di sicurezza.
- Affidabilità: il pilastro dell'affidabilità si concentra sulla capacità di prevenire e ripristinare rapidamente i guasti per soddisfare le esigenze aziendali e dei clienti. Gli argomenti chiave includono elementi fondamentali relativi alla configurazione, ai requisiti interprogettuali, alla pianificazione del ripristino e al modo in cui gestiamo il cambiamento.
- Efficienza delle prestazioni: il pilastro dell'efficienza delle prestazioni si concentra sull'uso efficiente delle risorse IT e informatiche. Gli argomenti chiave includono la selezione delle dimensioni e dei tipi corretti per le risorse in base ai requisiti del carico di lavoro, il monitoraggio delle prestazioni e il processo per prendere decisioni informate e mantenere l'efficienza man mano che le esigenze aziendali cambiano.
- Ottimizzazione dei costi: il pilastro dell'ottimizzazione dei costi si concentra sull'evitare costi non necessari. Gli argomenti chiave includono la comprensione e il controllo di dove vengono spesi i soldi, la selezione del numero più appropriato e corretto di tipi di risorse, l'analisi della spesa nel tempo e la scalabilità per soddisfare le esigenze aziendali senza spendere troppo.

- Sostenibilità: il pilastro della sostenibilità si concentra sulla capacità di migliorare continuamente gli impatti sulla sostenibilità riducendo il consumo di energia e aumentando l'efficienza in tutti i componenti di un carico di lavoro, massimizzando i benefici derivanti dalle risorse fornite e riducendo al minimo le risorse totali richieste.

AWS Well-Architected Framework: descrive AWS come consente ai clienti di valutare e migliorare le proprie architetture basate sul cloud e comprendere meglio l'impatto aziendale delle loro decisioni di progettazione. Affronta i principi generali di progettazione, nonché le migliori pratiche e linee guida specifiche in sei aree concettuali che AWS definiscono i pilastri del Well-Architected Framework.

Responsabilità a livello di applicazione e responsabilità a livello di infrastruttura in AMS

Utilizzando AMS, l'infrastruttura e tutto ciò di cui ha bisogno per la manutenzione e la crescita vengono mantenute da AMS. Tuttavia, tutto ciò di cui avete bisogno per line-of-business le applicazioni o le applicazioni dei prodotti, viene sviluppato, distribuito e gestito da voi.

Con l'aiuto di strumenti di distribuzione delle applicazioni, come CodeDeploy and, o Chef AWS CloudFormation, Puppet, Ansible o Saltstack, l'implementazione delle applicazioni nell'infrastruttura gestita da AMS può essere completamente automatizzata.

Per maggiori informazioni su cosa fa e cosa non fa AMS, consulta. Cosa facciamo, cosa non facciamo

Mutabilità delle EC2 istanze di Amazon in AMS

Tu e AMS potete mantenere le istanze Amazon Elastic Compute Cloud (Amazon EC2) nella vostra infrastruttura in due modi:

- Immutabile: questo modello utilizza Amazon Machine Images (AMIs) baked (creato) con le funzionalità necessarie. Quando si distribuisce un aggiornamento, le istanze esistenti vengono eliminate e completamente sostituite con nuove istanze create da un'AMI aggiornata. Per ridurre al minimo i tempi di inattività, questo processo progressivo lascia alcune istanze non aggiornate e accessibili mentre altre vengono aggiornate fino alla completa implementazione della nuova modifica.
- Mutabile: in questo modello, l'infrastruttura viene aggiornata con l'implementazione di nuovo codice sui sistemi esistenti nel cloud. Questo modello combina l'invio manuale degli aggiornamenti e

l'utilizzo infrastructure-as-code per la distribuzione degli aggiornamenti e non si basa su novità. AMIs

Questi modelli di manutenzione sono descritti più dettagliatamente nelle sezioni successive di questa guida.

Utilizzo di AWS Secrets Manager con risorse AMS

Esistono molti casi in cui potrebbe essere necessario condividere segreti con AMS, ad esempio:

- Reimpostazione della password principale per l'istanza RDS
- Certificati per sistemi di bilanciamento del carico
- Ottenere credenziali di lunga durata per gli utenti IAM da AMS

Il modo più sicuro per condividere informazioni riservate con AMS è tramite AWS Secrets Manager; segui questi passaggi:

1. Accedi alla AWS Console utilizzando il tuo accesso federato e il CustomerReadOnly ruolo per single-account landing zone (SALZ); usa uno di questi ruoli, AWSManaged ServicesSecurityOpsRole AWSManagedServicesAdminRole, e AWSManaged ServicesChangeManagementRole per la landing zone multi-account (MALZ).
2. Accedi alla console di gestione AWS Secrets e fai clic su Archivia un nuovo segreto.
3. Seleziona «Altro tipo di segreti».
4. Inserisci il valore segreto come testo semplice e fai clic su Avanti.
5. Immettete il nome e la descrizione del segreto. Il nome deve sempre iniziare con "customer-shared/*». Ad esempio "customer-shared/license-2018». Una volta terminato, continua facendo clic su Avanti.
6. Usa la crittografia KMS predefinita.
7. Lascia disabilitata la rotazione automatica e fai clic su Avanti.
8. Controlla e fai clic su Store per salvare il segreto.
9. Rispondi a noi in una richiesta di servizio AMS con il nome segreto e l'ARN, in modo che possiamo identificare e recuperare il segreto. Per informazioni sulla creazione di richieste di assistenza, consulta Esempi di richieste di assistenza.

Implementazione delle applicazioni in AMS

Durante l'onboarding, AWS Managed Services (AMS) collabora con te per determinare l'infrastruttura di cui hai bisogno.

L'infrastruttura di base include un cloud privato AWS virtuale (VPC), la sicurezza delle comunicazioni tramite un trust forestale ADFS, le sottoreti di base (DMZ, Shared Services e Private) rispecchiate su due zone di disponibilità e configurate con un NAT gestito, bastioni, sistemi di bilanciamento del carico pubblici (DX) e la sicurezza richiesta. AWS Direct Connect Le risorse delle applicazioni verranno distribuite nella sottorete privata o destinata alle applicazioni dei clienti. Per saperne di più su una tipica architettura AMS, consulta la AWS Managed Services User Guide.

L'infrastruttura che distribuisci, una volta completate le operazioni di base, dovrebbe includere tutti i componenti per le applicazioni e lo sviluppo delle applicazioni.

Funzionalità di distribuzione delle applicazioni in AMS

Alcuni dei modi in cui è possibile distribuire le applicazioni in AMS. Di seguito sono riportati i dettagli su ciascun metodo.

Esempi di funzionalità di distribuzione delle applicazioni

Nome del metodo	Distribuzione dell'infrastruttura	AMI o elemento/i chiave/i	Installazione dell'applicazione
Applicazioni mutabili, AMI AMS			
Distribuzione manuale delle applicazioni	CT completo o Tier and Tie CTs	AMI fornita da AMS	Invia Access management CT, installa l'applicazione manualmente.
UserData distribuzione dell'applicazione con agente applicativo (ad esempio Chef, Puppet, ecc.)			Utilizzate il provisioning CT con uno UserData script che installa un agente applicativo e che

Nome del metodo	Distribuzione dell'infrastruttura	AMI o elemento/i chiave/i	Installazione dell'applicazione
			script/agent installa l'applicazione.
UserData distribuzione di applicazioni senza agenti (ad esempio Ansible, Salt SSH, ecc.)			Invia Access Management CT, installa l'agente applicativo. Distribuisci l'applicazione con gli strumenti di distribuzione delle applicazioni.
Applicazioni mutabili, AMI personalizzate			
Implementazione di applicazioni AMI personalizzate (non ASG)	CT completo o Tier and Tie CTs	AMI personalizzata. AMS AMI -> personalizza con Application Deploy Tooling Agent -> crea EC2 istanza (CT) -> crea AMI (CT).	Application Deploy Tooling (ad esempio Chef), sfruttando gli agenti, implementa l'applicazione.
Distribuzione di applicazioni AWS Database Migration Service (DMS)	Sincronizzazione di AWS DMS con lo stack di database relazionali AMS esistente.	AMI personalizzata	Il cliente o il partner utilizza AWS Database Migration Service; AMS verifica i componenti AMS al momento del lancio

Nome del metodo	Distribuzione dell'infrastruttura	AMI o elemento/i chiave/i	Installazione dell'applicazione
Implementazione dell'applicazione Workload Ingest	Workload Ingest CT migrato dai partner instance/AMI e avviato dal cliente.		Il partner migra l'istanza, crea AMI nel VPC gestito da AMS del cliente; il cliente utilizza Workload Ingest CT per lanciare lo stack in AMS. Per informazioni dettagliate, vedi Inserimento del carico di lavoro AMS (WIGS).
Applicazioni immutabili			
Implementazione di applicazioni AMI personalizzate (ASG)	CT o Tier and Tie completo CTs	AMS AMI -> personalizza -> crea EC2 istanza (CT) -> crea AMI (CT) -> crea gruppo Auto Scaling.	Auto Scaling implementa l'applicazione con l'AMI personalizzata Per informazioni dettagliate, vedi Implementazioni di app Tier e Tie in AMS.
Applicazioni mutabili o immutabili			

Nome del metodo	Distribuzione dell'infrastruttura	AMI o elemento/i chiave/i	Installazione dell'applicazione
Distribuzione di applicazioni Template personalizzate CloudFormation	CloudFormation modello	CloudFormation Modello AWS -> customize/prepare per AMS -> Distribuzione | Ingestione | Stack da CloudFormation modello | Crea (ct-36cn2avfrrj9v).	AMS distribuisce l'applicazione sul tuo account utilizzando il modello personalizzato e convalida la distribuzione dell'applicazione. CloudFormation Per informazioni dettagliate, vedi Acquisizione di AMS CloudFormation .
Importazione di database SQL	Operazioni AMS (Altro | Altro CT)	Database SQL locale -> file.bak -> Database SQL AMS RDS -> Gestione | Altro | Altro | Crea (ct-1e1xtak34nx76) per l'importazione.	AMS importa il database locale nel database RDS gestito da AMS. Per informazioni dettagliate, vedi Importazione di database (DB) su AMS RDS per Microsoft SQL Server.
Servizio di migrazione del Database (DMS)	Operazioni AMS (multiple CTs)	Database locale -> istanza di replica DMS -> sottogruppo di replica DMS -> endpoint di destinazione DMS -> endpoint di origine DMS -> attività di replica DMS.	AMS importa il database locale nel database S3 o RDS di destinazione gestito da AMS. Per informazioni dettagliate, vedi AWS Database Migration Service (AWS DMS).

Nome del metodo	Distribuzione dell'infrastruttura	AMI o elemento/i chiave/i	Installazione dell'applicazione
CodeDeploy distribuzione delle applicazioni	CodeDeploy	Applicazione -> CodeDeploy applicazione -> gruppo CodeDeploy di distribuzione -> CodeDeploy distribuzione.	A seconda dell'utilizzo, della distribuzione sul posto o Blue/Green dell'applicazione. Per informazioni dettagliate, vedi CodeDeploy richieste.

Pianificazione della distribuzione delle applicazioni in AMS

Per una serie di domande consigliate a cui rispondere per abilitare la distribuzione delle applicazioni, consulta. Appendice: Questionario di onboarding delle applicazioni Le domande riguardano la descrizione di:

- Riepilogo della distribuzione
- Componenti di implementazione dell'infrastruttura
- Piattaforma di hosting delle applicazioni
- Modello di distribuzione delle applicazioni
- Dipendenze delle applicazioni
- Certificati SSL per applicazioni di prodotto

Inserimento del carico di lavoro AMS (WIGS)

Argomenti

- Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows
- In che modo la migrazione cambia la tua risorsa
- Migrazione dei carichi di lavoro: processo standard
- Migrazione dei carichi di lavoro: CloudEndure landing zone (SALZ)
- Account AMS Tools (migrazione dei carichi di lavoro)

- Migrazione dei carichi di lavoro: convalida pre-ingestione di Linux
- Migrazione dei carichi di lavoro: convalida pre-ingestione di Windows
- Workload Ingest Stack: creazione

Utilizza il workload ingest change type (CT) AMS con un partner di migrazione cloud AMS per spostare i carichi di lavoro esistenti in un VPC gestito da AMS. Utilizzando AMS workload ingest, puoi creare un'AMI AMS personalizzata dopo aver spostato le istanze migrate su AMS. Questa sezione descrive il processo, i prerequisiti e le fasi che il tuo partner di migrazione e tu adottiamo per l'acquisizione del carico di lavoro AMS.

> **Important**
>
> Il sistema operativo deve essere supportato da AMS workload ingest. Per i sistemi operativi supportati, vedere. Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows
> Ogni carico di lavoro e account è diverso. AMS collaborerà con te per prepararti a un risultato positivo.

Il diagramma seguente illustra il processo di inserimento del carico di lavoro AMS.

Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows

Prima di importare una copia di un'istanza locale in AWS Managed Services (AMS), devono essere soddisfatti alcuni prerequisiti. Questi sono i prerequisiti, inclusi quelli che differiscono tra i sistemi operativi Windows e Linux.

> **Note**
>
> Per semplificare il processo di determinazione se le istanze sono pronte per l'inserimento, sono stati creati strumenti di convalida per Windows e Linux. Questi strumenti possono essere scaricati ed eseguiti direttamente sui server locali e sulle EC2 istanze in AWS. Linux Pre-WIGS Validation.zip, Windows Pre-WIGS Validation.zip.

PRIMA DI INIZIARE, per Linux e Windows:

- Esegui una scansione antivirus completa.

- L'istanza deve avere il profilo dell'`customer-mc-ec2-instance-profile`istanza.
- Installa l'agente Amazon EC2 Systems Manager (SSM) e assicurati che l'agente SSM sia attivo e funzionante.
- Si consiglia un minimo di 10 GB di spazio libero su disco sul volume root per eseguire AMS workload ingest (WIGS). Dal punto di vista operativo, AMS consiglia un utilizzo del disco inferiore al 75% e avvisa quando l'utilizzo del disco raggiunge l'85%.
- Stabilite un periodo di tempo per l'inserimento con il vostro partner di migrazione.
- L'AMI personalizzato esiste come EC2 istanza nell'account AMS di produzione di destinazione (questa è la responsabilità del partner di migrazione).

Important

Il sistema operativo deve essere supportato da AMS workload ingest.

- Sono supportati i seguenti sistemi operativi:
  - Microsoft Windows Server: 2008 R2, 2012, 2012 R2, 2016, 2019 e 2022
  - Linux: Amazon Linux 2023, Amazon Linux 2 e Amazon Linux, CentOS 7.x, CentOS 6.5-6.10, Oracle Linux 7: versioni secondarie 7.5 e successive, Oracle Linux 8: versioni secondarie fino a 8.3, RHEL 8.x, RHEL 7.x, RHEL 6.5-6.10, SUSE Linux Enterprise Server 15 e versioni specifiche SAP, SUSE Linux Enterprise Server 12, Ubuntu 18.04 SP3 SP4 SP5
- Le seguenti non sono supportate: AMIs
  - AMI minima di Amazon Linux 2023.

Note

Gli endpoint AMS API/CLI (amscm e amsskms) si trovano nella regione AWS della Virginia settentrionale,. `us-east-1` A seconda di come è impostata l'autenticazione e della regione AWS in cui si trovano l'account e le risorse, potrebbe essere necessario aggiungerli `--region us-east-1` quando si emettono i comandi. Potrebbe anche essere necessario aggiungere`--profile saml`, se questo è il metodo di autenticazione utilizzato.

Prerequisiti LINUX

Rispettate i requisiti elencati in Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows e assicuratevi di quanto segue prima di inviare una RFC WIGS:

- Sono installati i driver di rete avanzati più recenti; vedi Enhanced Networking su Linux.
- I componenti software di terze parti in conflitto con i componenti AMS sono stati rimossi:
  - Client antivirus
  - Client di backup
  - Software di virtualizzazione (come VM Tools o servizi di integrazione Hyper-V)
  - Software di gestione degli accessi (come SSSD, Centrify o PBIS)
- Assicurati che SSH sia configurato correttamente: ciò abilita temporaneamente l'autenticazione con chiave privata per SSH. AMS lo utilizza con il nostro strumento di gestione della configurazione. Usa questi comandi:

```
sudo grep -q "^PubkeyAuthentication" /etc/ssh/sshd_config && sudo sed "s/
^PubkeyAuthentication=.*/PubkeyAuthentication yes/" -i /etc/ssh/sshd_config || sudo
 sed "$ a\PubkeyAuthentication yes" -i /etc/ssh/sshd_config
```

```
sudo grep -q "^AuthorizedKeysFile" /etc/ssh/sshd_config && sudo sed "s/
^AuthorizedKeysFile=.*/AuthorizedKeysFile %h\/.ssh\/authorized_keys/" -i /etc/ssh/
sshd_config || sudo sed "$ a\AuthorizedKeysFile %h/.ssh/authorized_keys" -i /etc/ssh/
sshd_config
```

- Assicurati che Yum sia configurato correttamente: RedHat richiede una licenza per utilizzare i loro repository Yum. L'istanza deve essere concessa in licenza tramite un Satellite Server o un Cloud Server. RedHat Utilizza uno di questi link se è necessaria la licenza:
  - Red Hat Satellite
  - Accesso al cloud Red Hat
- Se utilizzi Red Hat Satellite, WIGS richiede l'aggiunta di Red Hat Software Collections (RHSCL). Il sistema WIGS utilizza RHSCL per aggiungere un interprete Python3.6 a tutto ciò che è configurato sul sistema. Per supportare questa soluzione, devono essere disponibili i seguenti repository:
  - rhel-server-rhscl
  - rhel-server-releases-optional

Prerequisiti Windows

Rispettate i requisiti elencati in Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows e assicuratevi di quanto segue prima di inviare una RFC WIGS:

- È installata la versione 3 o superiore di Powershell.
- AWS EC2 Config viene installato sull'istanza con il carico di lavoro da migrare.
- Installa i driver AWS che supportano i tipi di istanze di ultima generazione: PV, ENA e NVMe. Puoi utilizzare le informazioni contenute in questi link:
  - Aggiornamento dei driver PV sulle istanze Windows
  - Rete avanzata su Windows
  - NVMe Driver AWS per istanze Windows
  - Parte 3: Aggiornamento dei driver AWS NVMe
  - Parte 5: installazione del driver di porta seriale per istanze bare metal
  - Parte 6: Aggiornamento delle impostazioni di gestione dell'alimentazione
- (Facoltativo ma consigliato) Disattiva i servizi critici delle applicazioni: imposta i servizi applicativi critici, come i database, su disabilitati, ma assicurati che tutte le modifiche siano documentate in modo che possano essere ripristinate alla modalità di avvio originale durante la fase di verifica dell'applicazione.
- (Facoltativo ma consigliato) Crea un AMI Failsafe dall'istanza preparata:
  - Usa la distribuzione | Componenti stack avanzati | AMI | Crea
  - Durante la creazione, aggiungi un tag Key=Name, value=application-ID_ IngestReady
  - Attendi la creazione dell'AMI prima di procedere
- I componenti software di terze parti in conflitto con i componenti AMS sono stati rimossi:
  - Client antivirus
  - Client di backup
  - Software di virtualizzazione (come VM Tools o servizi di integrazione Hyper-V)

> **Note**
>
> Il programma di End-of-Support migrazione per Windows server (EMP) include strumenti per migrare le applicazioni legacy da Windows Server 2003, 2008 e 2008 R2 a versioni più recenti e supportate su AWS, senza alcun refactoring.

In che modo la migrazione cambia la tua risorsa

La RFC di importazione descritta in questa sezione compie il passaggio successivo di aggiunta di configurazioni all'istanza, una volta migrata all'account AMS, in modo che AMS possa gestirla.

Le configurazioni aggiunte sono specifiche di AMS come segue.

Modifiche apportate alle istanze Linux importate:

- Software installato:
  - Cloud Init: utilizzato per configurare le chiavi private per Jarvis Access.
  - Python 3 (linguaggio di scripting) per tutti i sistemi operativi supportati (tranne CentOS 6, RHEL 8, 7). OracleLinux
  - AWS CloudFormation Python Helper Scripts: CloudFormation AWS fornisce script utilizzati per installare software e avviare servizi su istanze Amazon. EC2
  - AWS CLI: l'AWS CLI è uno strumento open source basato sull'SDK AWS per Python (Boto) che fornisce comandi per interagire con i servizi AWS.
  - Agente AWS SSM: l'agente SSM elabora le richieste dal servizio Systems Manager e configura la macchina come specificato nella richiesta.
  - AWS CloudWatch Logs Agent: invia i log a. CloudWatch
  - AWS CodeDeploy: un servizio di distribuzione che automatizza le distribuzioni di applicazioni su istanze Amazon, EC2 istanze locali o funzioni Lambda serverless.
  - Ruby: richiesto per CodeDeploy
  - Strumenti per le prestazioni del sistema (sysstat): Sysstat contiene varie utilità per monitorare le prestazioni del sistema e l'attività di utilizzo.
  - AD Bridge (precedentemente PowerBroker Identity Services): unisce gli host non Microsoft ai domini Active Directory.
  - Trend Micro Deep Security Agent: software antivirus.
- Software modificato:
  - Le istanze sono configurate per utilizzare il fuso orario UTC.

Modifiche apportate alle istanze di Windows importate:

- Software installato:

  - Strumenti AWS per Windows PowerShell: gli strumenti AWS PowerShell consentono a sviluppatori e amministratori di gestire i propri servizi e risorse AWS nell'ambiente di PowerShell scripting.
  - Trend Micro Deep Security Agent: protezione antivirus
  - PowerShell Moduli AMS contenenti PowerShell codice per il controllo di avvio, aggiunta ad Active Directory, monitoraggio, sicurezza e registrazione.
- Software modificato:
  - La versione 1 di Server Message Block (SMB) è disabilitata.
  - Windows Remote Management (WinRM) è abilitato e configurato per l'ascolto sulla porta 5986. Viene inoltre creata una regola firewall che consente questa porta in ingresso.
- Software che potrebbe essere installato o modificato:
  - Microsoft.Net Framework 4.5 (piattaforma per sviluppatori), se viene rilevata una versione inferiore a.Net Framework 4.5.
  - Per Windows 2012 e Windows 2012R2, eseguiamo l'aggiornamento alla versione 5.1. PowerShell

Migrazione dei carichi di lavoro: processo standard

> **Note**
>
> Poiché per questo processo sono necessarie due parti, questa sezione descrive le attività per ciascuna di esse: un AMS Cloud Migration Partner (partner di migrazione) e un proprietario dell'applicazione (tu).

1. Partner di migrazione, configurazione:

   a. Il partner di migrazione invia una richiesta di servizio ad AMS per un ruolo IAM allo scopo di migrare l'istanza. Per i dettagli sull'invio di richieste di servizio, consulta Esempi di richieste di assistenza.

   b. Il partner di migrazione invia una richiesta di accesso amministrativo. Il team AMS Operations fornisce al partner di migrazione l'accesso al tuo account tramite il ruolo IAM richiesto.

2. Partner di migrazione, Migrate Individual Workload:

   a. Il partner di migrazione migra la tua non AWS istanza verso una sottorete del tuo account AMS tramite Amazon nativo EC2 o altri strumenti di migrazione, con il profilo dell'istanza `customer-mc-ec2-instance-profile` IAM (deve essere presente nell'account).

   b. Il partner di migrazione invia una RFC con l'istanza migrata Deployment | Ingestion | Stack from migration partner | Create CT (ct-257p9zjk14ija); per dettagli sulla creazione e l'invio di questa RFC, consulta. Workload Ingest Stack: creazione

      L'output di esecuzione dell'RFC restituisce un ID di istanza, un indirizzo IP e un ID AMI.

      Il partner di migrazione ti fornisce l'ID dell'istanza del carico di lavoro creato nel tuo account.

3. Tu, accedi e convalida la migrazione:

   a. Utilizzando l'output di esecuzione fornito (ID AMI, ID istanza e indirizzo IP) dal partner di migrazione, invia una RFC di accesso e accedi allo stack AMS appena creato e verifica che l'applicazione funzioni correttamente. Per i dettagli, consulta Richiesta di accesso all'istanza.

   b. Se sei soddisfatto, puoi continuare a utilizzare l'istanza lanciata come stack a 1 livello e and/or utilizzare l'AMI per creare stack aggiuntivi, inclusi i gruppi di Auto Scaling.

   c. Se non sei soddisfatto della migrazione, invia una richiesta di servizio e fai riferimento allo stack e alla RFC IDs; AMS collaborerà con te per risolvere i tuoi dubbi.

CloudEndure Di seguito viene descritto il processo di acquisizione del carico di lavoro delle landing zone.

Migrazione dei carichi di lavoro: CloudEndure landing zone (SALZ)

Questa sezione fornisce informazioni sulla configurazione di una landing zone (SALZ) a migrazione intermedia per le istanze cutover CloudEndure (CE) da rendere disponibili per un workload ingest (WIGS) RFC.

Per ulteriori informazioni CloudEndure, CloudEndure consulta Migrazione.

> **Note**
>
> Si tratta di una LZ e di un modello di migrazione predefiniti e dotati di protezione avanzata.

Prerequisiti:

- Un account AMS per un cliente
- Integrazione di rete e accesso tra l'account AMS e il cliente in sede
- Un account CloudEndure
- Un flusso di lavoro di pre-approvazione per la revisione e l'approvazione di AMS Security, eseguito con CA and/or CSDM (ad esempio, l'uso improprio delle credenziali permanenti degli utenti IAM consente di attivare istanze e gruppi di sicurezza) create/delete

> **Note**
>
> I processi specifici di preparazione e migrazione sono descritti in questa sezione.

Preparazione: Tu e l'operatore AMS:

1. Prepara una richiesta di modifica (RFC) con Management | Altro | Altro | Aggiorna il tipo di modifica ad AMS per le seguenti risorse e aggiornamenti. Puoi inviare un altro aggiornamento RFCs | Altro | Altro o uno solo. Per i dettagli su tale RFC/CT, consulta Altro | Altro aggiornamento con queste richieste:

    a. Assegna un blocco CIDR secondario nel tuo VPC AMS; un blocco CIDR temporaneo che verrà rimosso una volta completata la migrazione. Assicurati che il blocco non entri in conflitto con nessun percorso esistente verso la tua rete locale. Ad esempio, se il tuo CIDR VPC AMS è 10.0.0.0/16 ed esiste un percorso di ritorno alla rete locale 10.1.0.0/16, il CIDR secondario temporaneo potrebbe essere 10.255.255.0/24. Per informazioni sui blocchi AWS CIDR, consulta VPC e Subnet Sizing.
    b. Crea una nuova sottorete privata all'interno del VPC AMS Initial-garden. Nome di esempio:. `migration-temp-subnet`
    c. Crea una nuova tabella di routing per la sottorete con solo route VPC e NAT (Internet) locali, per evitare conflitti con il server di origine durante il cutover dell'istanza e possibili interruzioni. Assicurati che il traffico in uscita verso Internet sia consentito per il download delle patch e che i prerequisiti di AMS WIGS possano essere scaricati e installati.
    d. Aggiorna il gruppo di sicurezza Managed AD per consentire il traffico in entrata e in uscita. to/from `migration-temp-subnet` Richiedi inoltre che il tuo gruppo di sicurezza EPS

Load Balancer (ELB) (ad esempio:`mc-eps-McEpsElbPrivateSecurityGroup-M790XBZEEX74`) venga aggiornato per consentire la nuova sottorete privata (ad esempio). `migration-temp-subnet` Se il traffico proveniente dalla sottorete dedicata CloudEndure (CE) non è consentito su tutte e tre le porte TCP, l'ingestione di WIGS avrà esito negativo.

e. Infine, richiedi una nuova CloudEndure policy IAM e un nuovo utente IAM. <Customer Application Subnet (s) + Temp Migration Subnet>La policy richiede il tuo numero di account corretto e la sottorete IDs nella `RunInstances` dichiarazione dovrebbe essere: your.

   Per visualizzare una CloudEndure policy IAM preapprovata da AMS: decomprimi il file WIGS Cloud Endure Landing Zone Example e apri il. `customer_cloud_endure_policy.json`

   > **Note**
   >
   > Se desideri una politica più permissiva, discuti con il tuo personale di cosa hai bisogno e richiedi, se necessario, un'AMS Security Review CloudArchitect/CSDM e l'approvazione prima di inviare una RFC che implementi la policy.

2. Le fasi di preparazione da utilizzare CloudEndure per l'acquisizione del carico di lavoro AMS sono state completate e, se il partner di migrazione ha completato le fasi di preparazione, la migrazione è pronta per essere eseguita. La RFC WIGS viene inviata dal partner di migrazione.

> **Note**
>
> Le chiavi utente IAM non verranno condivise direttamente, ma devono essere digitate nella console di CloudEndure gestione dall'operatore AMS durante una sessione di condivisione dello schermo.

Preparazione: Partner di migrazione e operatore AMS:

1. Crea un progetto di CloudEndure migrazione.

   a. Durante la creazione del progetto, fai in modo che AMS inserisca le credenziali utente IAM nelle sessioni di condivisione dello schermo.

   b. In Impostazioni di replica -> Scegli la sottorete in cui verranno lanciati i server di replica, seleziona sottorete. customer-application-x

c. In Impostazioni di replica -> Scegli i gruppi di sicurezza da applicare ai server di replica, seleziona entrambi i gruppi di sicurezza Sentinel (solo privati e). EgressAll

2. Definite le opzioni di cutover per le macchine (istanze).

   a. Sottorete:. migration-temp-subnet

   b. Gruppo di sicurezza: entrambi i gruppi di sicurezza «Sentinel» (solo privato e). EgressAll

      Le istanze cutover devono essere in grado di comunicare con AMS Managed AD e con gli endpoint pubblici di AWS.

   c. IP elastico: nessuno

   d. IP pubblico: no

   e. Ruolo IAM: profilo a customer-mc-ec2 istanze

      Il ruolo IAM deve consentire la comunicazione SSM. Meglio usare l'impostazione predefinita di AMS.

   f. Imposta i tag secondo la convenzione.

Migrazione: Partner per la migrazione:

1. Crea uno stack fittizio su AMS. Utilizzi l'ID dello stack per accedere ai bastioni.
2. Installa l'agente CloudEndure (CE) sul server di origine. Per i dettagli, vedere Installazione degli agenti.
3. Crea credenziali di amministratore locale sul server di origine.
4. Pianifica una breve finestra di modifica e fai clic su Cutover, quando sei pronto. Ciò finalizza la migrazione e reindirizza gli utenti alla regione AWS di destinazione.
5. Richiedi l'accesso dell'amministratore dello stack allo stack fittizio, vedi Admin Access Request.
6. Accedi al bastion, quindi all'istanza cutover utilizzando le credenziali di amministratore locale che hai creato.
7. Crea un'AMI a prova di errore. Per i dettagli sulla creazione AMIs, consulta AMI Create.
8. Preparare l'istanza per l'ingestione, vedi. Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows
9. Esegui WIGS RFC sull'istanza, vedi. Workload Ingest Stack: creazione

Account AMS Tools (migrazione dei carichi di lavoro)

Il tuo account Multi-Account Landing Zone Tools (con VPC) aiuta ad accelerare le operazioni di migrazione, aumenta la tua posizione di sicurezza, riduce costi e complessità e standardizza il modello di utilizzo.

Un account Tools offre quanto segue:

- Un limite ben definito per l'accesso alle istanze di replica per gli integratori di sistema al di fuori dei carichi di lavoro di produzione.
- Consente di creare una camera isolata per verificare la presenza di malware o percorsi di rete sconosciuti in un carico di lavoro prima di trasferirlo in un account con altri carichi di lavoro.
- Essendo una configurazione definita dell'account, offre tempi più rapidi per l'onboarding e la configurazione per la migrazione dei carichi di lavoro.
- Percorsi di rete isolati per proteggere il traffico proveniente dall'account locale -> -> Tools account CloudEndure -> AMS ingerita dall'immagine. Una volta che l'immagine è stata inserita, puoi condividerla con l'account di destinazione tramite una RFC AMS Management | Advanced stack components | AMI | Share (ct-1eiczxw8ihc18).

Diagramma di architettura di alto livello:

Usa Deployment | Managed landing zone | Management account | Create tools account (con VPC) change type (ct-2j7q1hgf26x5c), per implementare rapidamente un account di strumenti e creare un'istanza di un processo di inserimento del carico di lavoro in un ambiente Multi-Account Landing Zone. Vedi Account di gestione, Account Tools: Creazione (con VPC).

> **Note**
>
> Ti consigliamo di avere due zone di disponibilità (AZs), poiché si tratta di un hub di migrazione.
> Per impostazione predefinita, AMS crea i seguenti due gruppi di sicurezza (SGs) in ogni account. Conferma che questi due SGs siano presenti. Se non sono presenti, apri una nuova richiesta di assistenza con il team AMS per richiederli.
>
> - SentinelDefaultSecurityGroupPrivateOnlyEgressAll
> - InitialGarden-SentinelDefaultSecurityGroupPrivateOnly

Assicurati che le istanze di CloudEndure replica vengano create nella sottorete privata in cui sono presenti percorsi di ritorno all'ambiente locale. Puoi confermarlo assicurandoti che le tabelle di routing per la sottorete privata abbiano una route predefinita per tornare a TGW. Tuttavia, se si esegue un cut over CloudEndure automatico, si dovrebbe passare alla sottorete privata «isolata», dove non esiste alcuna via di ritorno alla rete locale, ma è consentito solo il traffico Internet in uscita. È fondamentale garantire che si verifichi il cutover nella sottorete isolata per evitare potenziali problemi alle risorse locali.

Prerequisiti:

1. Livello di supporto Plus o Premium.
2. L'account dell'applicazione IDs per la chiave KMS in cui AMIs vengono distribuite.
3. L'account degli strumenti, creato come descritto in precedenza.

AWS Servizio di migrazione delle applicazioni (AWS MGN)

AWS Application Migration Service (AWS MGN) può essere utilizzato nel tuo account MALZ Tools tramite il ruolo `AWSManagedServicesMigrationRole` IAM che viene creato automaticamente durante il provisioning dell'account Tools. È possibile utilizzare AWS MGN per migrare applicazioni e database eseguiti su versioni supportate dei sistemi operativi Windows e Linux.

Per la maggior parte delle up-to-date informazioni sull' Regione AWS assistenza, consulta l'elenco dei servizi AWS regionali.

Se il tuo preferito non Regione AWS è attualmente supportato da AWS MGN o il sistema operativo su cui vengono eseguite le tue applicazioni non è attualmente supportato da AWS MGN, prendi in considerazione l'utilizzo di CloudEndure Migration nel tuo account Tools.

Richiedere l'inizializzazione AWS di MGN

AWS MGN deve essere inizializzato da AMS prima del primo utilizzo. Per richiederlo per un nuovo account Tools, invia una richiesta RFC Management | Other | Other dall'account Tools con questi dettagli:

```
RFC Subject=Please initialize AWS MGN in this account
```

```
RFC Comment=Please click 'Get started' on the MGN welcome page here:

    https://console.aws.amazon.com/mgn/home?region=MALZ_PRIMARY_REGION#/welcome using
 all default values
    to 'Create template' and complete the initialization process.
```

Una volta che AMS avrà completato con successo la RFC e inizializzato AWS MGN nel tuo account Tools, potrai utilizzarlo `AWSManagedServicesMigrationRole` per modificare il modello predefinito in base alle tue esigenze.

Abilita l'accesso al nuovo account AMS Tools

Una volta creato l'account Tools, AMS ti fornisce un ID account. Il passaggio successivo consiste nel configurare l'accesso al nuovo account. Segui questi passaggi.

1. Aggiorna i gruppi Active Directory appropriati con l'account appropriato IDs.

   Ai nuovi account creati con AMS vengono assegnati i criteri relativi ai ReadOnly ruoli e un ruolo che consente agli utenti di archiviare i file. RFCs

   L'account Tools dispone anche di un ruolo e di un utente IAM aggiuntivi:

   - Ruolo IAM: `AWSManagedServicesMigrationRole`
   - Utente IAM: `customer_cloud_endure_user`
2. Richiedi politiche e ruoli per consentire ai membri del team di integrazione dei servizi di configurare il livello successivo di strumenti.

   Vai alla console AMS e archivia quanto segue RFCs:

   a. Crea una chiave KMS. Usa Crea chiave KMS (auto) o Crea chiave KMS (revisione richiesta).

      Poiché utilizzi KMS per crittografare le risorse acquisite, l'utilizzo di una singola chiave KMS condivisa con il resto degli account dell'applicazione Multi-Account Landing Zone garantisce la sicurezza delle immagini importate, in modo che possano essere decrittografate nell'account di destinazione.

   b. Condividi la chiave KMS.

      Utilizza Management | Advanced stack components | KMS key | Share (revisione richiesta) change type (ct-05yb337abq3x5) per richiedere che la nuova chiave KMS venga condivisa con gli account dell'applicazione in cui risiederà quella acquisita. AMIs

Grafico di esempio della configurazione finale dell'account:

Esempio di politica IAM CloudEndure preapprovata da AMS

Per visualizzare una CloudEndure policy IAM preapprovata da AMS: decomprimi il file di esempio WIGS Cloud Endure Landing Zone e apri il. `customer_cloud_endure_policy.json`

Test della connettività e della end-to-end configurazione dell'account AMS Tools

1. Inizia con la configurazione CloudEndure e l'installazione dell' CloudEndure agente su un server che si replicherà su AMS.
2. Crea un progetto in. CloudEndure
3. Inserisci le AWS credenziali condivise quando hai eseguito i prerequisiti, tramite Secrets Manager.
4. Nelle impostazioni di replica:
   a. Seleziona entrambi i gruppi di sicurezza AMS «Sentinel» (solo privati e EgressAll) per l'opzione Scegli i gruppi di sicurezza da applicare ai server di replica.
   b. Definite le opzioni di cutover per le macchine (istanze). Per informazioni, consultate la Fase 5. Tagliare
   c. Sottorete: sottorete privata.
5. Gruppo di sicurezza:
   a. Seleziona entrambi i gruppi di sicurezza AMS «Sentinel» (solo privato e EgressAll).
   b. Le istanze Cutover devono comunicare con l'Active Directory (MAD) gestito da AMS e con gli endpoint pubblici: AWS
      i. IP elastico: nessuno
      ii. IP pubblico: no
      iii. Ruolo IAM: profilo a customer-mc-ec 2 istanze
   c. Imposta i tag secondo la tua convenzione di etichettatura interna.
6. Installa l' CloudEndure agente sulla macchina e cerca l'istanza di replica da visualizzare nel tuo account AMS nella EC2 console.

Il processo di ingestione di AMS:

Igiene dell'account AMS Tools

Ti consigliamo di ripulire l'account dopo aver condiviso l'AMI e non avrai più bisogno delle istanze replicate:

- Dopo l'ingestione dell'istanza WIGs :
  - Istanza Cutover: come minimo, interrompi o termina questa istanza, una volta completato il lavoro, tramite la console AWS
  - Backup AMI pre-incorporazione: rimuovi una volta che l'istanza è stata inserita e l'istanza locale terminata
  - Istanze inserite da AMS: disattivate lo stack o terminate una volta che l'AMI è stata condivisa
  - AMS-Ingested: elimina una volta completata la condivisione con l'account di destinazione AMIs
- Pulizia alla fine della migrazione: documenta le risorse distribuite tramite la modalità Sviluppatore per garantire che la pulizia avvenga regolarmente, ad esempio:
  - Gruppi di sicurezza
  - Risorse create tramite Cloud-formation
  - Rete ACK
  - Sottorete
  - VPC
  - Tabella di routing
  - Ruoli
  - Utenti e account

Migrazione su larga scala - Migration Factory

Vedi Presentazione della soluzione AWS CloudEndure Migration Factory.

Migrazione dei carichi di lavoro: convalida pre-ingestione di Linux

Puoi verificare che l'istanza sia pronta per l'inserimento nel tuo account AMS. La convalida pre-ingestione del carico di lavoro (WIGS) esegue controlli quali il tipo di sistema operativo, lo spazio disponibile su disco, l'esistenza di software di terze parti in conflitto, ecc. Una volta eseguita, la convalida pre-ingestione di WIGS produce una tabella su schermo, con un file di registro opzionale. I risultati forniscono pass/fail lo stato di ogni controllo di convalida insieme al motivo di eventuali errori. Inoltre, è possibile personalizzare i test di convalida in base alle proprie esigenze.

Domande frequenti:

- Come posso utilizzare la convalida pre-ingestione di Linux WIGS?

  Segui questi passaggi per scaricare e utilizzare gli script di convalida pre-ingestione di AMS Linux WIGS:

  1. Scarica un file ZIP con gli script di convalida

     File zip di convalida pre-ingestione di Linux WIGS.

  2. Decomprimi le regole allegate in una cartella a tua scelta.
  3. Segui le istruzioni nel file readme.md.

- Quali convalide vengono eseguite dalla convalida pre-ingestione di Linux WIGS?

  La soluzione di convalida pre-ingestione di AMS Linux WIGS convalida quanto segue:

  1. Ci sono almeno 5 Gigabyte liberi nel volume di avvio.
  2. Il sistema operativo è supportato da AMS.
  3. L'istanza ha un profilo di istanza specifico.
  4. L'istanza non contiene software antivirus o software di virtualizzazione.
  5. SSH è configurato correttamente.
  6. L'istanza ha accesso agli Yum Repositories.
  7. Sono installati driver di rete avanzati.
  8. L'istanza ha l'agente SSM ed è in esecuzione.

- Perché è disponibile il supporto per un file di configurazione personalizzato?

  Gli script sono progettati per essere eseguiti sia su server fisici locali che su istanze AWS. EC2 Tuttavia, come mostrato nell'elenco precedente, alcuni test falliranno se eseguiti in locale. Ad esempio, un server fisico in un datacenter non avrebbe un profilo di istanza. In casi come questi, puoi modificare il file di configurazione per saltare il test del profilo dell'istanza per evitare confusione.

- Come posso assicurarmi di avere la versione più recente dello script?

  Una up-to-date versione della soluzione Linux WIGS Pre-Ingestion Validation sarà disponibile nella sezione AMS Helper Files nella pagina principale della documentazione.

- Lo script è di sola lettura?

Lo script è progettato per essere di sola lettura ad eccezione dei file di registro che produce, ma è necessario seguire le migliori pratiche per eseguire lo script in un ambiente non di produzione.

- La convalida pre-ingestione di WIGS è disponibile per Windows?

  Sì. È disponibile nella sezione AMS Helper Files nella pagina principale della documentazione.

Migrazione dei carichi di lavoro: convalida pre-ingestione di Windows

Puoi utilizzare lo script di WIGs pre-validazione per verificare che l'istanza sia pronta per l'inserimento nel tuo account AMS. La convalida pre-ingestione del carico di lavoro (WIGS) esegue controlli quali il tipo di sistema operativo, lo spazio disponibile su disco, l'esistenza di software di terze parti in conflitto e così via. Quando viene eseguita, la convalida pre-ingestione di WIGS produce una tabella su schermo e un file di registro opzionale. I risultati forniscono uno pass/fail stato per ogni controllo di convalida insieme al motivo dell'errore. Inoltre, è possibile personalizzare i test di convalida.

Domande frequenti:

- Come si utilizza la convalida pre-ingestione di Windows WIGS?

  È possibile eseguire la convalida da una GUI e da un browser Web oppure utilizzare Windows PowerShell, SSM Run Command o SSM Session Manager.

  Opzione 1: Esegui da una GUI e da un browser web

  Per eseguire la WIGs pre-convalida di Windows da una GUI e da un browser Web, procedi come segue:

  1. Scarica un file ZIP con gli script di convalida:

     File ZIP di convalida pre-ingestione di Windows WIGS.
  2. Decomprimi le regole allegate in una cartella a tua scelta.
  3. Segui le istruzioni contenute nel file README.md.

  Opzione 2: Esegui da Windows PowerShell, SSM Run Command o SSM Session Manager

  Windows 2016 e versioni successive

  1. Scarica il file ZIP con gli script di convalida.

```
$DestinationFile = "$env:TEMP\WIGValidation.zip"
```

```
$Bucket = 'https://docs.aws.amazon.com/managedservices/latest/appguide/samples/
windows-prewigs-validation.zip'
$DestinationFile = "$env:TEMP\WIGValidation.zip"
$ScriptFolder = "$env:TEMP\AWSManagedServices.PreWigs.Validation"
```

2. Rimuovi i file esistenti da. `C:\Users\AppData\Local\Temp\AWSManagedServices.PreWigs.Validation`

```
Remove-Item $scriptFolder -Recurse -Force -ErrorAction Ignore
```

3. Invoca lo script.

```
Invoke-WebRequest -Uri $bucket -OutFile $DestinationFile
Add-Type -Assembly "system.io.compression.filesystem"
```

4. Decomprimi i file allegati in una cartella a tua scelta.

```
[io.compression.zipfile]::ExtractToDirectory($DestinationFile, $env:TEMP)
```

5. Esegui lo script di convalida in modo interattivo e visualizza i risultati.

```
Import-Module .\AWSManagedServices.PreWigs.Validation.psm1 -force
Invoke-PreWIGsValidation -RunWithoutExitCodes
```

6. (Facoltativo) Per acquisire i codici di errore elencati nella sezione Codici di uscita, esegui lo script senza l'`RunWithoutExitCodes`opzione. Nota che questo comando termina la PowerShell sessione attiva.

```
Import-Module .\AWSManagedServices.PreWigs.Validation.psm1 -force
Invoke-PreWIGsValidation
```

Windows 2012 R2 e versioni precedenti

Se utilizzi Windows Server 2012R2 o versioni precedenti, devi impostare TLS prima di scaricare il file zip. Per impostare TLS, completa i seguenti passaggi:

1. Scarica il file ZIP con gli script di convalida.

```
$DestinationFile = "$env:TEMP\WIGValidation.zip"
```

```
$Bucket = 'https://docs.aws.amazon.com/managedservices/latest/appguide/samples/
windows-prewigs-validation.zip'
$DestinationFile = "$env:TEMP\WIGValidation.zip"
$ScriptFolder = "$env:TEMP\AWSManagedServices.PreWigs.Validation"
```

2. Se sono presenti file di convalida esistenti, rimuovili.

```
Remove-Item $scriptFolder -Recurse -Force -ErrorAction Ignore
```

3. Imposta la versione TLS.

```
[System.Net.ServicePointManager]::SecurityProtocol = 'TLS12'
```

4. Scarica la convalida WIG.

```
Invoke-WebRequest -Uri $bucket -OutFile $DestinationFile
Add-Type -Assembly "system.io.compression.filesystem"
```

5. Decomprimi le regole allegate in una cartella a tua scelta.

```
[io.compression.zipfile]::ExtractToDirectory($DestinationFile, $env:TEMP)
```

6. Esegui lo script di convalida in modo interattivo e visualizza i risultati.

```
Import-Module .\AWSManagedServices.PreWigs.Validation.psm1 -force
Invoke-PreWIGsValidation -RunWithoutExitCodes
```

7. (Facoltativo) Per acquisire i codici di errore elencati nella sezione Codici di uscita, esegui lo script senza l' RunWithoutExitCodes opzione. Nota che questo comando termina la PowerShell sessione attiva.

```
Import-Module .\AWSManagedServices.PreWigs.Validation.psm1 -force
Invoke-PreWIGsValidation
```

> **Note**
>
> È possibile scaricare ed eseguire gli PowerShell script. Per fare ciò, scarica il file pre-wigs-validation-powershell-scripts.zip.

- Quali convalide vengono eseguite dalla convalida pre-ingestione di Windows WIGS?

La soluzione di convalida pre-ingestione di AMS Windows WIGS convalida quanto segue:

1. Sul volume di avvio sono disponibili almeno 10 Gigabyte.
2. Il sistema operativo è supportato da AMS.
3. L'istanza ha un profilo di istanza specifico.
4. L'istanza non contiene software antivirus o software di virtualizzazione.
5. Il DHCP è abilitato su almeno una scheda di rete.
6. L'istanza è pronta per Sysprep.
   - Per 2008 R2 e 2012 Base e R2, Sysprep verifica che:
     - Esiste un file unattend.xml
     - Il file sppnp.dll (se presente) non è danneggiato
     - Il sistema operativo non è stato aggiornato
     - Sysprep non è stato eseguito più del numero massimo di volte previsto dalle linee guida Microsoft
   - Per il 2016 e versioni successive, tutti i controlli precedenti vengono ignorati in quanto nessuno dei due causa problemi per quel sistema operativo
7. Il sottosistema di strumentazione di gestione Windows (WMI) è integro.
8. I driver richiesti sono installati.
9. L'agente SSM è installato e funzionante.
10. Viene fornito un avviso per verificare se la macchina è in periodo di prova a causa della configurazione della licenza RDS.
11. Le chiavi di registro richieste sono impostate correttamente. Per ulteriori dettagli, consultate il file README nel file zip di convalida pre-ingestione.

- Perché è disponibile il supporto per un file di configurazione personalizzato?

  Gli script sono progettati per essere eseguiti sia su server fisici locali che su istanze AWS. EC2 Tuttavia, come mostrato nell'elenco precedente, alcuni test falliranno se eseguiti in locale. Ad esempio, un server fisico in un datacenter non avrebbe un profilo di istanza. In casi come questi, puoi modificare il file di configurazione per saltare il test del profilo dell'istanza per evitare confusione.

- Come posso assicurarmi di avere la versione più recente dello script?

  Una up-to-date versione della soluzione di convalida pre-ingestione di Windows WIGS sarà disponibile nella sezione AMS Helper Files nella pagina principale della documentazione.

- Lo script è di sola lettura?

  Lo script è progettato per essere di sola lettura ad eccezione dei file di registro che produce, ma è necessario seguire le migliori pratiche per eseguire lo script in un ambiente non di produzione.

- La convalida pre-ingestione di WIGS è disponibile per Linux?

  Sì. La versione Linux è stata lanciata il 31 ottobre 2019. È disponibile nella sezione AMS Helper Files nella pagina principale della documentazione.

Workload Ingest Stack: creazione

Migrazione di un'istanza in uno stack AMS con la console

Schermata di questo tipo di modifica nella console AMS:

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un Oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.

4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

> **Note**
>
> Se la RFC viene rifiutata, l'output di esecuzione include un collegamento ai CloudWatch log di Amazon. Gli AMS Workload Ingest (WIGS) RFCs vengono rifiutati quando i requisiti non sono soddisfatti, ad esempio se sull'istanza viene rilevato un software antivirus. I CloudWatch log includeranno informazioni sul requisito non soddisfatto e sulle azioni da intraprendere per porvi rimedio.

Migrazione di un'istanza verso uno stack AMS con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

È possibile utilizzare l'AMS CLI per creare un'istanza AMS da un'istanza non AMS migrata a un account AMS.

> **Note**
>
> Assicurati di aver rispettato i prerequisiti; consulta Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows.

Per verificare la versione del tipo di modifica, utilizzate questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws amscm create-rfc --change-type-id "ct-257p9zjk14ija" --change-type-version "2.0" --
title "AMS-WIG-TEST-NO-ACTION" --execution-parameters "{\"InstanceId\":\"INSTANCE_ID\",
\"TargetVpcId\":\"VPC_ID\",\"TargetSubnetId\":\"SUBNET_ID\",\"TargetInstanceType\":
\"t2.large\",\"ApplyInstanceValidation\":true,\"Name\":\"WIG-TEST\",\"Description\":
\"WIG-TEST\",\"EnforceIMDSV2\":\"false\"}"
```

CREAZIONE DEL MODELLO:

1. 0emette lo schema JSON dei parametri di esecuzione per questo tipo di modifica in un file; l'esempio lo chiama .json: MigrateStackParams

```
aws amscm get-change-type-version --change-type-id "ct-257p9zjk14ija" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > MigrateStackParams.json
```

2. Modifica e salva il file JSON dei parametri di esecuzione. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"InstanceId":          "MIGRATED_INSTANCE_ID",
"TargetVpcId":         "VPC_ID",
"TargetSubnetId":      "SUBNET_ID",
"Name":                "Migrated-Stack",
"Description":         "Create-Migrated-Stack",
"EnforceIMDSV2":       "false"
}
```

3. Esporta il file JSON del modello RFC; l'esempio lo MigrateStackRfc chiama .json:

```
aws amscm create-rfc --generate-cli-skeleton > MigrateStackRfc.json
```

4. Modifica e salva il file.json. MigrateStackRfc Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeId":         "ct-257p9zjk14ija",
"ChangeTypeVersion":    "2.0",
"Title":                "Migrate-Stack-RFC"
}
```

5. Crea la RFC, specificando il MigrateStackRfc file e il MigrateStackParams file:

```
aws amscm create-rfc --cli-input-json file://MigrateStackRfc.json  --execution-
parameters file://MigrateStackParams.json
```

   Nella risposta ricevi l'ID della nuova RFC e puoi utilizzarlo per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

   La nuova istanza viene visualizzata nell'elenco delle istanze dell'account del proprietario dell'applicazione per il VPC pertinente.

6. Una volta completata correttamente la RFC, invia una notifica al proprietario dell'applicazione in modo che possa accedere alla nuova istanza e verificare che il carico di lavoro sia operativo.

> **Note**
>
> Se la RFC viene rifiutata, l'output di esecuzione include un collegamento ai CloudWatch log di Amazon. Gli AMS Workload Ingest (WIGS) RFCs vengono rifiutati quando i requisiti non sono soddisfatti, ad esempio se sull'istanza viene rilevato un software antivirus. I CloudWatch log includeranno informazioni sul requisito non soddisfatto e sulle azioni da intraprendere per porvi rimedio.

Suggerimenti

> **Note**
>
> Assicurati di aver rispettato i prerequisiti; vedi Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows.

> **Note**
>
> Se un tag sull'istanza da migrare ha la stessa chiave di un tag fornito nella RFC, la RFC fallisce.

> **Note**
>
> Puoi specificare fino a quattro zone di destinazione IDs, porte e disponibilità.

> **Note**
>
> Se la RFC viene rifiutata, l'output di esecuzione include un collegamento ai CloudWatch log di Amazon. Gli AMS Workload Ingest (WIGS) RFCs vengono rifiutati quando i requisiti non sono soddisfatti, ad esempio se sull'istanza viene rilevato un software antivirus. I CloudWatch

log includeranno informazioni sul requisito non soddisfatto e sulle azioni da intraprendere per porvi rimedio.

> **Note**
>
> Se la RFC viene rifiutata, l'output di esecuzione include un collegamento ai CloudWatch log di Amazon. Gli AMS Workload Ingest (WIGS) RFCs vengono rifiutati quando i requisiti non sono soddisfatti, ad esempio se sull'istanza viene rilevato un software antivirus. I CloudWatch log includeranno informazioni sul requisito non soddisfatto e sulle azioni da intraprendere per porvi rimedio.

Se necessario, vedere Errore di ingestione del carico di lavoro (WIGS).

Acquisizione di AMS CloudFormation

L'AMS AWS CloudFormation ingest change type (CT) consente di utilizzare i CloudFormation modelli esistenti, con alcune modifiche, per distribuire stack personalizzati in un VPC gestito da AMS.

Argomenti

- AWS CloudFormation Linee guida, best practice e limitazioni per l'inserimento
- AWS CloudFormation Ingest: esempi
- Crea uno stack di importazione CloudFormation
- Aggiorna lo stack di importazione AWS CloudFormation
- CloudFormation Approva un set di modifiche allo stack di importazione
- Update AWS CloudFormation stack: protezione dalla terminazione
- Implementazioni IAM automatizzate che utilizzano CFN ingest o stack update in AMS CTs

Il processo di acquisizione di AMS prevede quanto segue: AWS CloudFormation

- Prepara e carica il tuo CloudFormation modello personalizzato in un bucket S3 oppure fornisci il modello in linea durante la creazione della RFC. Se utilizzi un bucket S3 con un URL predefinito; per ulteriori informazioni, consulta presign.

- Invia il tipo di modifica CloudFormation da importare ad AMS in un RFC. Per la procedura dettagliata sul tipo di modifica del tipo di importazione di CFN, consulta. Crea uno stack di importazione CloudFormation Per esempi di CFN ingest, vedi. AWS CloudFormation Ingest: esempi
- Una volta creato lo stack, puoi aggiornarlo e rimediare ai suoi errori; inoltre, se l'aggiornamento dovesse fallire, puoi approvarlo e implementarlo in modo esplicito. Tutte queste procedure sono descritte in questa sezione.

  Per informazioni sul rilevamento della deriva mediante CFN, vedere New — CloudFormation Drift Detection.

> **Note**
>
> - Questo tipo di modifica ha ora una versione 2.0. La versione 2.0 è automatizzata, non eseguita manualmente. Ciò consente all'esecuzione CT di procedere più rapidamente. Con questa versione vengono introdotti due nuovi parametri: CloudFormationTemplate, che consente di incollare un CloudFormation modello personalizzato nella RFC e VpcId, che consente di utilizzare l' CloudFormation ingest con la landing zone multi-account AMS.
> - La versione 1.0 è un tipo di modifica manuale. Ciò significa che un operatore AMS deve intraprendere alcune azioni prima che il tipo di modifica possa concludersi con successo. È richiesta almeno una revisione. Questa versione richiede inoltre che il valore del parametro CloudFormationTemplateS3Endpoint sia un URL prefirmato.

AWS CloudFormation Linee guida, best practice e limitazioni per l'inserimento

Per consentire ad AMS di elaborare il CloudFormation modello, esistono alcune linee guida e restrizioni.

Linee guida

Per ridurre AWS CloudFormation gli errori AWS CloudFormation durante l'acquisizione, segui queste linee guida:

- Non incorporare credenziali o altre informazioni riservate nel modello: il CloudFormation modello è visibile nella AWS CloudFormation console, quindi non è necessario incorporare credenziali o dati sensibili nel modello. Il modello non può contenere informazioni sensibili. Le seguenti risorse sono consentite solo se utilizzi AWS Secrets Manager per il valore:
  - `AWS::RDS::DBInstance` - [MasterUserPassword,TdeCredentialPassword]
  - `AWS::RDS::DBCluster` - [MasterUserPassword]
  - `AWS::ElastiCache::ReplicationGroup` - [AuthToken]

  > **Note**
  >
  > Per informazioni sull'utilizzo di un segreto di AWS Secrets Manager in una proprietà di risorsa, consulta Come creare e recuperare segreti gestiti in AWS Secrets Manager utilizzando CloudFormation modelli AWS e Utilizzo di riferimenti dinamici per specificare i valori dei modelli.

- Usa gli snapshot di Amazon RDS per creare istanze DB RDS: in questo modo eviti di dover fornire un. MasterUserPassword
- Se il modello inviato contiene un profilo di istanza IAM, deve avere il prefisso «cliente». Ad esempio, l'utilizzo di un profilo di istanza con il nome 'example-instance-profile' causa un errore. Utilizzate invece un profilo di istanza con il nome 'customer-example-instance-profile'.
- Non includere dati sensibili in **`AWS::EC2::Instance`** - [UserData]. UserData non deve contenere password, chiavi API o altri dati sensibili. Questo tipo di dati può essere crittografato e archiviato in un bucket S3 e scaricato sull'istanza utilizzando. UserData
- La creazione di policy IAM tramite CloudFormation modelli è supportata con alcuni vincoli: le policy IAM devono essere riviste e approvate da AMS. SecOps Attualmente supportiamo solo l'implementazione di ruoli IAM con policy in linea che contengono autorizzazioni preapprovate. In altri casi, le policy IAM non possono essere create utilizzando CloudFormation modelli perché ciò avrebbe la precedenza sul processo AMS. SecOps
- SSH KeyPairs non è supportato: è necessario accedere EC2 alle istanze Amazon tramite il sistema di gestione degli accessi AMS. Il processo AMS RFC ti autentica. Non puoi includere coppie di chiavi SSH nei CloudFormation modelli perché non disponi delle autorizzazioni per creare coppie di chiavi SSH e sovrascrivere il modello di gestione degli accessi AMS.
- Le regole di accesso ai gruppi di sicurezza sono limitate: non è possibile avere un intervallo CIDR di origine compreso tra 0.0.0.0/0 o uno spazio di indirizzi indirizzabile pubblicamente, con una porta TCP diversa da 80 o 443.

- Segui le AWS CloudFormation linee guida quando scrivi i modelli di CloudFormation risorse: assicurati di utilizzare il type/property nome di dati corretto per la risorsa facendo riferimento alla Guida per l'utente di quella risorsa.AWS CloudFormation Ad esempio, il tipo di dati della SecurityGroupIds proprietà in una AWS::EC2::Instance risorsa è 'Elenco di valori String', quindi ["sg-aaaaaaaa"] è ok (con parentesi), ma «sg-aaaaaaaa» no (senza parentesi).

  Per ulteriori informazioni, consulta AWS Resource and Property Types Reference.
- Configura i tuoi CloudFormation modelli personalizzati per utilizzare i parametri definiti in AMS CloudFormation ingest CT — Quando configuri il tuo CloudFormation modello per utilizzare i parametri definiti in AMS CloudFormation ingest CT, puoi riutilizzare il CloudFormation modello per creare stack simili inviandolo con i valori dei parametri modificati nell'input CT con Management | Custom stack | Stack from CloudFormation template | Update CT (ct-361tlo1k7339x). Per vedere un esempio, consulta AWS CloudFormation Esempi di inserimento: definizione delle risorse.
- Gli endpoint del bucket Amazon S3 con un URL predefinito non possono essere scaduti: se utilizzi un endpoint bucket Amazon S3 con un URL predefinito, verifica che l'URL Amazon S3 prefirmato non sia scaduto. Una CloudFormation richiesta RFC di importazione inviata con un URL del bucket Amazon S3 prefirmato scaduto viene rifiutata.
- Wait Condition richiede una logica di segnale: Wait Condition viene utilizzata per coordinare la creazione di risorse dello stack con azioni di configurazione esterne alla creazione dello stack. Se utilizzi la risorsa Wait Condition nel modello, AWS CloudFormation attende un segnale di successo e contrassegna la creazione dello stack come un errore se non viene emesso il numero di segnali di successo. È necessario disporre di una logica per il segnale se si utilizza la risorsa Wait Condition. Per ulteriori informazioni, vedere Creazione di condizioni di attesa in un modello.

Best practice

Di seguito sono riportate alcune best practice che è possibile utilizzare per migrare le risorse utilizzando il processo di acquisizione di AMS AWS CloudFormation :

- Invia IAM e altre risorse relative alle politiche in un unico CT: se puoi utilizzare strumenti automatizzati CTs come CloudFormation Ingest per implementare ruoli IAM, ti consigliamo di farlo. In altri casi, AMS consiglia di raccogliere tutte le risorse IAM o altre risorse relative alle policy e di inviarle in un unico tipo di gestione | Altro | Altro | Crea modifica (ct-1e1xtak34nx76). Ad esempio, combina tutti i ruoli IAM necessari, i profili di EC2 istanza IAM Amazon, gli aggiornamenti delle policy IAM per i ruoli IAM esistenti, le policy dei bucket Amazon S3, le politiche Amazon SNS/ Amazon SQS e così via, e invia un RFC ct-1e1xtak34nx76 in modo che queste risorse preesistenti

possano essere semplicemente referenziate all'interno dei futuri modelli di acquisizione. CloudFormation

- EC2 le istanze vengono avviate e aggiunte correttamente al dominio: questa operazione viene eseguita automaticamente come best practice. Per garantire che le EC2 istanze Amazon lanciate tramite uno stack di CloudFormation ingest vengano avviate e si aggiungano correttamente al dominio, AMS include una risorsa di gruppo CreationPolicy e una per un UpdatePolicy Auto Scaling (ovvero, se queste policy non esistono già).
- È necessario specificare il parametro dell'istanza database Amazon RDS: quando si crea un database Amazon RDS tramite AWS CloudFormation ingest, è necessario specificare il `DBSnapshotIdentifier` parametro per eseguire il ripristino da uno snapshot DB precedente. Questo è necessario perché attualmente l' AWS CloudFormation ingest non gestisce dati sensibili.

Per un esempio di come utilizzare un CloudFormation modello per l'acquisizione di CloudFormation modelli AMS, vedi. AWS CloudFormation Ingest: esempi

Convalida del modello

Puoi convalidare automaticamente il CloudFormation modello prima di inviarlo ad AMS.

I modelli inviati ad AMS AWS CloudFormation ingest sono convalidati per garantire che possano essere implementati in sicurezza all'interno di un account AMS. Il processo di convalida verifica quanto segue:

- Risorse supportate: vengono utilizzate solo risorse AWS CloudFormation supportate da AMS ingest. Per ulteriori informazioni, consulta Risorse supportate.
- Supportata AMIs : l'AMI nel modello è un'AMI supportata da AMS. Per informazioni su AMS AMIs, vedere. Immagini di macchine AMS Amazon (AMIs)
- Sottorete AMS Shared Services: il modello non tenta di avviare risorse nella sottorete AMS Shared Services.
- Politiche relative alle risorse: non esistono politiche in materia di risorse eccessivamente permissive, come una policy sui bucket S3 leggibile o scrivibile pubblicamente. AMS non consente l'accesso a bucket S3 leggibili o scrivibili pubblicamente. Account AWS

AWS CloudFormation Convalida con Linter

Puoi convalidare automaticamente il CloudFormation modello prima di inviarlo ad AMS utilizzando lo strumento Linter. AWS CloudFormation

Lo strumento AWS CloudFormation Linter è il modo migliore per convalidare il CloudFormation modello in quanto fornisce la convalida di resource/property nomi, tipi di dati e funzioni. Per ulteriori informazioni, consulta aws-cloudformation/. cfn-python-lint

L'output AWS CloudFormation Linter del modello mostrato in precedenza è il seguente:

```
$ cfn-lint -t ./testtmpl.json
E3002 Invalid Property Resources/SNSTopic/Properties/Name
./testtmpl.json:6:9
```

Per facilitare la convalida offline dei CloudFormation modelli, AMS ha sviluppato una serie di regole di convalida personalizzate collegabili per lo strumento Linter. AWS CloudFormation Si trovano nella pagina Risorse per gli sviluppatori della console AMS.

Segui questi passaggi per utilizzare gli script di AWS CloudFormation convalida prima dell'ingestione:

1. Installa lo strumento Linter. AWS CloudFormation Per le istruzioni di installazione, vedi aws-cloudformation /cfn-lint.
2. Scarica un file.zip con script di convalida:

   Regole personalizzate CFN Lint.
3. Decomprimi le regole allegate in una cartella a tua scelta.
4. Convalida il CloudFormation modello eseguendo il seguente comando:

   ```
   cfn-lint --template {TEMPLATE_FILE} --append-rules {DIRECTORY_WITH_CUSTOM_RULES}
   ```

CloudFormation ingest stack: esempi di validatori CFN

Questi esempi possono aiutarti a preparare il tuo modello per un'importazione di successo.

Convalida del formato

Verifica che il modello contenga una sezione «Risorse» e che tutte le risorse in essa definite abbiano un valore «Tipo».

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Description" : "Create a SNS topic",
  "Resources": {
    "SnsTopic": {
```

```
      "Type": "AWS::SNS::Topic"
    }
  }
}
```

Verifica che le chiavi principali del modello siano consentite. Le chiavi principali consentite sono:

```
[
  "AWSTemplateFormatVersion",
  "Description",
  "Mappings",
  "Parameters",
  "Conditions",
  "Resources",
  "Rules",
  "Outputs",
  "Metadata"
]
```

La revisione manuale richiede la convalida

Se il modello contiene le seguenti risorse, la convalida automatica fallisce e avrai bisogno di una revisione manuale.

Le politiche mostrate sono aree ad alto rischio dal punto di vista della sicurezza. Ad esempio, una policy sui bucket S3 che consente a chiunque, ad eccezione di utenti o gruppi specifici, di creare oggetti o autorizzazioni di scrittura è estremamente pericolosa. Pertanto convalidiamo le politiche e le approviamo o neghiamo in base ai contenuti, e tali politiche non possono essere create automaticamente. Stiamo esaminando possibili approcci per risolvere questo problema.

Al momento non disponiamo di una convalida automatica delle seguenti risorse.

```
[
    "S3::BucketPolicy",
    "SNS::TopicPolicy",
    "SQS::QueuePolicy"
]
```

Convalida dei parametri

Convalidalo se a un parametro del modello non viene fornito un valore; deve avere un valore predefinito.

Convalida degli attributi delle risorse

Controllo degli attributi obbligatorio: determinati attributi devono esistere per determinati tipi di risorse.

- "VPCOptions" deve esistere in `AWS::OpenSearch::Domain`
- "CludsterSubnetGroupName" deve esistere in `AWS::Redshift::Cluster`

```
{
    "AWS::OpenSearch::Domain": [
      "VPCOptions"
    ],
    "AWS::Redshift::Cluster": [
      "ClusterSubnetGroupName"
    ]
}
```

Controllo degli attributi non consentiti: alcuni attributi devono *non* esistere per determinati tipi di risorse.

- "SecretString" non deve esistere in "» AWS::SecretsManager::Secret
- "MongoDbSettings" non deve esistere in "AWS::DMS::Endpoint»

```
{
  "AWS::SecretsManager::Secret": [
    "SecretString"
  ],
  "AWS::DMS::Endpoint": [
    "MongoDbSettings"
  ]
}
```

Controllo dei parametri SSM: per gli attributi nell'elenco seguente, i valori devono essere specificati tramite Secrets Manager o Systems Manager Parameter Store (Secure String Parameter):

```
{
  "RDS::DBInstance": [
    "MasterUserPassword",
    "TdeCredentialPassword"
  ],
```

```
  "RDS::DBCluster": [
    "MasterUserPassword"
  ],
  "ElastiCache::ReplicationGroup": [
    "AuthToken"
  ],
  "DMS::Certificate": [
    "CertificatePem",
    "CertificateWallet"
  ],
  "DMS::Endpoint": [
    "Password"
  ],
  "CodePipeline::Webhook": {
    "AuthenticationConfiguration": [
        "SecretToken"
    ]
  },
  "DocDB::DBCluster": [
    "MasterUserPassword"
  ]
},
```

Alcuni attributi devono rispettare determinati modelli; ad esempio, i nomi dei profili delle istanze IAM non devono iniziare con prefissi riservati AMS e il valore dell'attributo deve corrispondere all'espressione regolare specifica, come mostrato:

```
{
    "AWS::EC2::Instance": {
      "IamInstanceProfile": [
        "^(?!arn:aws:iam|ams|Ams|AMS|AWSManagedServices|Managed_Services|mc|Mc|MC|
sentinel|Sentinel).+",
        "arn:aws:iam::(\\$\\{AWS::AccountId\\}|[0-9]+):instance-profile/(?!ams|Ams|AMS|
AWSManagedServices|Managed_Services|mc|Mc|MC|sentinel|Sentinel).+"
      ]
    },
    "AWS::AutoScaling::LaunchConfiguration": {
      "IamInstanceProfile": [
        "^(?!arn:aws:iam|ams|Ams|AMS|AWSManagedServices|Managed_Services|mc|Mc|MC|
sentinel|Sentinel).+",
        "arn:aws:iam::(\\$\\{AWS::AccountId\\}|[0-9]+):instance-profile/(?!ams|Ams|AMS|
AWSManagedServices|Managed_Services|mc|Mc|MC|sentinel|Sentinel).+"
      ]
```

```
    },
    "AWS::EC2::LaunchTemplate": {
      "LaunchTemplateData.IamInstanceProfile.Name": [
        "^(?!ams|Ams|AMS|AWSManagedServices|Managed_Services|mc|Mc|MC|sentinel|
Sentinel).+"
      ],
      "LaunchTemplateData.IamInstanceProfile.Arn": [
        "arn:aws:iam::(\\$\\{AWS::AccountId\\}|[0-9]+):instance-profile\/(?!ams|Ams|
AMS|AWSManagedServices|Managed_Services|mc|Mc|MC|sentinel|Sentinel).+"
      ]
    }
}
```

Convalida delle risorse

Nel modello possono essere specificate solo le risorse consentite; tali risorse sono descritte in. Risorse supportate

EC2 gli stack e i gruppi Auto Scaling ASGs () non sono consentiti nello stesso stack a causa delle limitazioni di applicazione delle patch.

Convalida delle regole di ingresso dei gruppi di sicurezza

- Per le richieste che provengono dai tipi di modifica CFN Ingest Create o Stack Update CT:
  - Se (`IpProtocol`è tcp o 6) AND (Port è 80 o 443), non ci sono restrizioni sul valore `CidrIP`
  - Altrimenti, non `CidrIP` può essere 0.0.0.0/0
- Per le richieste che provengono da Service Catalog (prodotti Service Catalog):
  - Oltre alla convalida del tipo di modifica CFN Ingest Create o Stack Update CT, è `ip_protocols` possibile accedere alla porta `management_ports` con il protocollo in ingresso solo tramite: `allowed_cidrs`

```
{
      "ip_protocols": ["tcp", "6", "udp", "17"],
      "management_ports": [22, 23, 389, 636, 1494, 1604, 2222, 3389, 5900, 5901,
 5985, 5986],
      "allowed_cidrs": ["10.0.0.0/8", "100.64.0.0/10", "172.16.0.0/12",
 "192.168.0.0/16"]
  }
```

Limitazioni

Le seguenti caratteristiche e funzionalità attualmente non sono supportate dal processo di acquisizione AWS CloudFormation di AMS.

- YAML: non supportato. Sono supportati solo i modelli basati su JSON CloudFormation.
- Stack annidati: invece, progetta l'infrastruttura applicativa in modo da utilizzare un unico modello. Oppure, in alternativa, puoi utilizzare il riferimento cross-stack per separare le risorse su più stack in cui una risorsa dipende da un'altra. Per ulteriori informazioni, consulta la procedura dettagliata: consulta Resource Outputs in Another AWS Stack. CloudFormation
- CloudFormation set di stack: non supportato, a causa di implicazioni di sicurezza.
- Creazione di risorse IAM tramite CloudFormation modelli: sono supportati solo i ruoli IAM, a causa delle implicazioni sulla sicurezza.
- Dati sensibili: non supportati. Non includere dati sensibili nel modello o nei valori dei parametri. Se è necessario fare riferimento a dati sensibili, utilizzare Secrets Manager per archiviare e recuperare questi valori. Per informazioni sull'utilizzo dei segreti di AWS Secrets Managers in una proprietà di risorsa, consulta Come creare e recuperare segreti gestiti in AWS Secrets Manager utilizzando CloudFormation modelli AWS e Utilizzo di riferimenti dinamici per specificare i valori dei modelli.

Risorse supportate

Le seguenti risorse AWS sono supportate nel processo di acquisizione AWS CloudFormation di AMS.

CloudFormation Ingest Stack: risorse supportate

Il sistema operativo dell'istanza deve essere supportato dall'inserimento del carico di lavoro AMS. Sono supportate solo le risorse AWS elencate qui.

- Gateway Amazon API
  - AWS::ApiGateway::Account
  - AWS::ApiGateway::ApiKey
  - AWS::ApiGateway::Authorizer
  - AWS::ApiGateway::BasePathMappatura
  - AWS::ApiGateway::ClientCertificate
  - AWS::ApiGateway::Deployment

- AWS::ApiGateway::DocumentationPart
- AWS::ApiGateway::DocumentationVersion
- AWS::ApiGateway::DomainName
- AWS::ApiGateway::GatewayResponse
- AWS::ApiGateway::Method
- AWS::ApiGateway::Model
- AWS::ApiGateway::RequestValidator
- AWS::ApiGateway::Resource
- AWS::ApiGateway::RestApi
- AWS::ApiGateway::Stage
- AWS::ApiGateway::UsagePlan
- AWS::ApiGateway::UsagePlanChiave
- AWS::ApiGateway::VpcLink

- Amazon API Gateway V2
  - AWS::ApiGatewayV2::Api
  - AWS::ApiGatewayV2::ApiGatewayManagedOverrides
  - AWS::ApiGatewayV2::ApiMapping
  - AWS::ApiGatewayV2::Authorizer
  - AWS::ApiGatewayV2::Deployment
  - AWS::ApiGatewayV2::DomainName
  - AWS::ApiGatewayV2::Integration
  - AWS::ApiGatewayV2::IntegrationResponse
  - AWS::ApiGatewayV2::Model
  - AWS::ApiGatewayV2::Route
  - AWS::ApiGatewayV2::RouteResponse
  - AWS::ApiGatewayV2::Stage
  - AWS::ApiGatewayV2::VpcLink
- AWS AppSync
  - AWS::AppSync::ApiCache
  - AWS::AppSync::ApiKey

  - AWS::AppSync::DataSource
  - AWS::AppSync::FunctionConfiguration
  - AWS::AppSync::GraphQLApi
  - AWS::AppSync::GraphQLSchema
  - AWS::AppSync::Resolver
- Amazon Athena
  - AWS::Athena::NamedQuery
  - AWS::Athena::WorkGroup
- AWS Backup
  - AWS::Backup::BackupVault
- Amazon CloudFront
  - AWS::CloudFront::Distribution
  - AWS::CloudFront::CloudFrontOriginAccessIdentity
  - AWS::CloudFront::StreamingDistribution
- Amazon CloudWatch
  - AWS::CloudWatch::Alarm
  - AWS::CloudWatch::AnomalyDetector
  - AWS::CloudWatch::CompositeAlarm
  - AWS::CloudWatch::Dashboard
  - AWS::CloudWatch::InsightRule
- CloudWatch Registri Amazon
  - AWS::Logs::LogGroup
  - AWS::Logs::LogStream
  - AWS::Logs::MetricFilter
  - AWS::Logs::SubscriptionFilter
- Amazon Cognito
  - AWS::Cognito::IdentityPool
  - AWS::Cognito::IdentityPoolRoleAttachment
  - AWS::Cognito::UserPool

  - AWS::Cognito::UserPoolClient

  - AWS::Cognito::UserPoolDominio
  - AWS::Cognito::UserPoolGruppo
  - AWS::Cognito::UserPoolIdentityProvider
  - AWS::Cognito::UserPoolResourceServer
  - AWS::Cognito::UserPoolRiskConfigurationAttachment
  - AWS::Cognito::UserPoolUICustomizationAllegato
  - AWS::Cognito::UserPoolUser
  - AWS::Cognito::UserPoolUserToGroupAttachment
- Amazon DocumentDB
  - AWS::DocDB:: DBCluster
  - AWS::DocDB:: DBCluster ParameterGroup
  - AWS::DocDB:: DBInstance
  - AWS::DocDB:: DBSubnet Gruppo
- Amazon DynamoDB
  - AWS::DynamoDB::Table
- Amazon EC2
  - AWS::EC2::Volume
  - AWS::EC2::VolumeAttachment
  - AWS::EC2::Instance
  - AWS:EC2: :EIP
  - AWS:EC2:: EIPAssociation
  - AWS::EC2::NetworkInterface
  - AWS::EC2::NetworkInterfaceAllegato
  - AWS::EC2::SecurityGroup
  - AWS::EC2::SecurityGroupIngresso
  - AWS::EC2::SecurityGroupUscita
  - AWS::EC2::LaunchTemplate
- AWS Batch
  - AWS::Batch::ComputeEnvironment
  - AWS::Batch::JobDefinition

  - AWS::Batch::JobQueue
- Amazon Elastic Container Registry (ECR)
  - AWS::ECR::Repository
- Amazon Elastic Container Service (ECS) (Fargate)
  - AWS::ECS::CapacityProvider
  - AWS::ECS::Cluster
  - AWS::ECS::PrimaryTaskImpostare
  - AWS::ECS::Service
  - AWS::ECS::TaskDefinition
  - AWS::ECS::TaskSet
- Amazon Elastic File System (EFS)
  - AWS::EFS::FileSystem
  - AWS::EFS::MountTarget
- Amazon ElastiCache
  - AWS::ElastiCache::CacheCluster
  - AWS::ElastiCache::ParameterGroup
  - AWS::ElastiCache::ReplicationGroup
  - AWS::ElastiCache::SecurityGroup
  - AWS::ElastiCache::SecurityGroupIngresso
  - AWS::ElastiCache::SubnetGroup
- Amazon EventBridge
  - AWS::Events::EventBus
  - AWS::Events::EventBusPolitica
  - AWS::Events::Rule
- Amazon FSx
  - AWS::FSx::FileSystem
- Amazon Inspector
  - AWS::Inspector::AssessmentTarget
  - AWS::Inspector::AssessmentTemplate
  - AWS::Inspector::ResourceGroup

- Amazon Kinesis Data Analytics
  - AWS::KinesisAnalytics::Application
  - AWS::KinesisAnalytics::ApplicationOutput
  - AWS::KinesisAnalytics::ApplicationReferenceDataSource
- Amazon Kinesis Data Firehose
  - AWS::KinesisFirehose::DeliveryStream
- Flusso di dati Amazon Kinesis
  - AWS::Kinesis::Stream
  - AWS::Kinesis::StreamConsumer
- Amazon MQ
  - AWS::AmazonMQ::Broker
  - AWS::AmazonMQ::Configuration
  - AWS::AmazonMQ::ConfigurationAssociation
- Amazon OpenSearch
  - AWS::OpenSearchService::Domain
- Amazon Relational Database Service (RDS)
  - AWS: :RDS:: DBCluster
  - AWS: :RDS:: DBCluster ParameterGroup
  - AWS: :RDS:: DBInstance
  - AWS: :RDS:: Gruppo DBParameter
  - AWS: :RDS:: Gruppo DBSubnet
  - AWS::RDS::EventSubscription
  - AWS::RDS::OptionGroup
- Amazon Route 53
  - AWS::Route53::HealthCheck
  - AWS::Route53::HostedZone
  - AWS::Route53::RecordSet
  - AWS::Route53::RecordSetGruppo
  - AWS::Route53Resolver::ResolverRule

  - AWS::Route53Resolver::ResolverRuleAssociazione

- Amazon S3
  - AWS::S3::Bucket
- Amazon SageMaker
  - AWS::SageMaker::CodeRepository
  - AWS::SageMaker::Endpoint
  - AWS::SageMaker::EndpointConfig
  - AWS::SageMaker::Model
  - AWS::SageMaker::NotebookInstance
  - AWS::SageMaker::NotebookInstanceLifecycleConfig
  - AWS::SageMaker::Workteam
- Amazon Simple Email Service (SES)
  - AWS::SES::ConfigurationSet
  - AWS::SES::ConfigurationSetEventDestination
  - AWS::SES::ReceiptFilter
  - AWS::SES::ReceiptRule
  - AWS::SES::ReceiptRuleImpostare
  - AWS::SES::Template
- Amazon SimpleDB
  - AWS::SDB::Domain
- Amazon SNS
  - AWS::SNS::Subscription
  - AWS::SNS::Topic
- Amazon SQS
  - AWS::SQS::Queue
- Amazon WorkSpaces
  - AWS::WorkSpaces::Workspace
- Applicazione AutoScaling
  - AWS::ApplicationAutoScaling::ScalableTarget
  - AWS::ApplicationAutoScaling::ScalingPolicy
- Amazon EC2 AutoScaling

  - AWS::AutoScaling::AutoScalingGruppo
  - AWS::AutoScaling::LaunchConfiguration
  - AWS::AutoScaling::LifecycleHook
  - AWS::AutoScaling::ScalingPolicy
  - AWS::AutoScaling::ScheduledAction
- AWS Certificate Manager
  - AWS::CertificateManager::Certificate
- AWS CloudFormation
  - AWS::CloudFormation::CustomResource
  - AWS::CloudFormation::Designer
  - AWS::CloudFormation::WaitCondition
  - AWS::CloudFormation::WaitConditionManiglia
- AWS CodeBuild
  - AWS::CodeBuild::Project
  - AWS::CodeBuild::ReportGroup
  - AWS::CodeBuild::SourceCredential
- AWS CodeCommit
  - AWS::CodeCommit::Repository
- AWS CodeDeploy
  - AWS::CodeDeploy::Application
  - AWS::CodeDeploy::DeploymentConfig
  - AWS::CodeDeploy::DeploymentGroup
- AWS CodePipeline
  - AWS::CodePipeline::CustomActionTipo
  - AWS::CodePipeline::Pipeline
  - AWS::CodePipeline::Webhook
- AWS Database Migration Service (DMS)
  - AWS::DMS::Certificate
  - AWS::DMS::Endpoint
  - AWS::DMS::EventSubscription

- AWS::DMS::ReplicationInstance
- AWS::DMS::ReplicationSubnetGruppo
- AWS::DMS::ReplicationTask

La MongoDbSettings proprietà nella AWS::DMS::Endpoint risorsa non è consentita.

Le seguenti proprietà sono consentite solo se risolte da AWS Secrets Manager: CertificatePem CertificateWallet proprietà nella AWS::DMS::Certificate risorsa e proprietà Password nella AWS::DMS::Endpoint risorsa.

- AWS Elastic Load Balancing - Application Load Balancer /Network Load Balancer
  - AWS::ElasticLoadBalancingV2::Listener
  - AWS::ElasticLoadBalancingV2::ListenerCertificate
  - AWS::ElasticLoadBalancingV2::ListenerRule
  - AWS::ElasticLoadBalancingV2::LoadBalancer
  - AWS::ElasticLoadBalancingV2::TargetGroup
- AWS Elastic Load Balancing - Classic Load Balancer
  - AWS::ElasticLoadBalancing::LoadBalancer
- AWS Elemental MediaConvert
  - AWS::MediaConvert::JobTemplate
  - AWS::MediaConvert::Preset
  - AWS::MediaConvert::Queue
- AWS Elemental MediaStore
  - AWS::MediaStore::Container
- AWS Identity and Access Management (IAM)
  - AWS::IAM::Role
- Streaming gestito da AWS per Apache Kafka (MSK)
  - AWS::MSK::Cluster
- AWS Glue
  - AWS::Glue::Classifier
  - AWS::Glue::Connection
  - AWS::Glue::Crawler
  - AWS::Glue::Database

  - AWS::Glue::DataCatalogEncryptionSettings
  - AWS::Glue::DevEndpoint
  - AWS::Glue::Job
  - AWS::Glue::MLTransform
  - AWS::Glue::Partition
  - AWS::Glue::SecurityConfiguration
  - AWS::Glue::Table
  - AWS::Glue::Trigger
  - AWS::Glue::Workflow
- AWS Key Management Service (KMS)
  - AWS::KMS::Key
  - AWS::KMS::Alias
- AWS Lake Formation
  - AWS::LakeFormation::DataLakeImpostazioni
  - AWS::LakeFormation::Permissions
  - AWS::LakeFormation::Resource
- AWS Lambda
  - AWS::Lambda::Alias
  - AWS::Lambda::EventInvokeConfig
  - AWS::Lambda::EventSourceMappatura
  - AWS::Lambda::Function
  - AWS::Lambda::LayerVersion
  - AWS::Lambda::LayerVersionAutorizzazione
  - AWS::Lambda::Permission
  - AWS::Lambda::Version
- Amazon Redshift
  - AWS::Redshift::Cluster
  - AWS::Redshift::ClusterParameterGruppo
  - AWS::Redshift::ClusterSubnetGruppo

- AWS Secrets Manager

  - AWS::SecretsManager::ResourcePolicy
  - AWS::SecretsManager::RotationSchedule
  - AWS::SecretsManager::Secret
  - AWS::SecretsManager::SecretTargetAllegato
- AWS Security Hub
  - AWS::SecurityHub::Hub
- AWS Step Functions
  - AWS::StepFunctions::Activity
  - AWS::StepFunctions::StateMachine
- AWS Systems Manager (SSM)
  - AWS::SSM::Parameter
- Amazon CloudWatch Synthetics
  - AWS::Synthetics::Canary
- Famiglia AWS Transfer Family
  - AWS::Transfer::Server
  - AWS::Transfer::User
- AWS WAF
  - AWS::WAF::ByteMatchImpostare
  - AWS: :WAF: IPSet
  - AWS::WAF::Rule
  - AWS::WAF::SizeConstraintImpostare
  - AWS::WAF::SqlInjectionMatchSet
  - AWS::WAF::WebACL
  - AWS::WAF::XssMatchImpostare
- AWS WAF Regionale
  - AWS::WAFRegional::ByteMatchImpostare
  - AWS::WAFRegional::GeoMatchImpostare
  - AWS:WAFRegional:: IPSet
  - AWS::WAFRegional::RateBasedRegola
  - AWS::WAFRegional::RegexPatternImpostare

  - AWS::WAFRegional::Rule
  - AWS::WAFRegional::SizeConstraintImpostare
  - AWS::WAFRegional::SqlInjectionMatchSet
  - AWS::WAFRegional::WebACL
  - AWS::WAFRegional::WebACLAssociation
  - AWS::WAFRegional::XssMatchImpostare
- AWS WAFv2
  - AWS:WAFv2:: IPSet
  - AWS::WAFv2::RegexPatternImpostare
  - AWS::WAFv2::RuleGroup
  - AWS::WAFv2::WebACL
  - AWS::WAFv2::WebACLAssociation

AWS CloudFormation Ingest: esempi

Di seguito sono riportati alcuni esempi dettagliati di come utilizzare lo stack Create con il tipo di modifica CloudFormation del modello.

Per scaricare un set di CloudFormation modelli di esempio per Regione AWS, consulta Modelli di esempio.

Per informazioni di riferimento sulle AWS CloudFormation risorse, consulta AWS Resource and Property Types Reference. Tuttavia, AMS supporta un set di risorse più piccolo, descritto inAcquisizione di AMS CloudFormation .

> **Note**
>
> AMS consiglia di raccogliere tutte le risorse IAM o altre risorse relative alle politiche e di inviarle in un unico tipo di gestione | Altro | Altro | Create change (ct-1e1xtak34nx76). Ad esempio, combina tutti i ruoli IAM necessari, i profili di istanza IAM, gli aggiornamenti delle policy IAM per i ruoli IAM esistenti, le policy dei bucket S3, SNS/SQS le politiche e così via, quindi invia un RFC ct-1e1xtak34nx76 in modo che queste risorse preesistenti possano essere referenziate all'interno dei futuri modelli CFN Ingest.

Argomenti

AWS CloudFormation Esempi di inserimento: definizione delle risorse

Quando si utilizza AMS AWS CloudFormation ingest, si personalizza un CloudFormation modello e lo si invia ad AMS in una RFC con il tipo di modifica di importazione (CloudFormation ct-36cn2avfrrj9v). Per creare un CloudFormation modello che possa essere riutilizzato più volte, aggiungete i parametri di configurazione dello stack all'input di esecuzione del tipo di modifica di importazione anziché codificarli nel modello. CloudFormation CloudFormation Il vantaggio principale è che puoi riutilizzare il modello.

Lo schema CloudFormation di input del tipo di modifica di AMS consente di scegliere fino a sessanta parametri in un CloudFormation modello e di fornire valori personalizzati.

Questo esempio mostra come definire una proprietà di risorsa, che può essere utilizzata in una varietà di CloudFormation modelli, come parametro in AMS CloudFormation ingest CT. Gli esempi in questa sezione mostrano in particolare l'utilizzo degli argomenti SNS.

Argomenti

Esempio 1: codifica rigida della proprietà della AWS CloudFormation SNSTopic risorsa **`TopicName`**

In questo esempio, si codifica fisicamente la `TopicName` proprietà della AWS CloudFormation SNSTopic risorsa nel CloudFormation modello. Notate che la `Parameters` sezione è vuota.

Per disporre di un CloudFormation modello che consenta di modificare il valore del SNSTopic nome di un nuovo stack senza dover creare un nuovo CloudFormation modello, è possibile utilizzare la

`Parameters` sezione AMS del tipo CloudFormation ingest change per effettuare tale configurazione. In questo modo, utilizzerai lo stesso CloudFormation modello in un secondo momento per creare un nuovo stack con un nome diverso. `SNSTopic`

```
{
  "AWSTemplateFormatVersion" : "2010-09-09",
  "Description" : "My SNS Topic",
  "Parameters" : {
  },
  "Resources" : {
    "SNSTopic" : {
      "Type" : "AWS::SNS::Topic",
      "Properties" : {
        "TopicName" : "MyTopicName"
      }
    }
  }
}
```

Esempio 2: utilizzare una SNSTopic risorsa per fare riferimento a un parametro nel tipo di modifica AMS

In questo esempio, si utilizza una `TopicName` proprietà di `SNSTopic` risorsa definita nel CloudFormation modello per fare riferimento a `Parameter` nel tipo di modifica AMS.

```
{
  "AWSTemplateFormatVersion" : "2010-09-09",
  "Description" : "My SNS Topic",
  "Parameters" : {
    "TopicName" : {
      "Type" : "String",
      "Description" : "Topic ID",
      "Default" : "MyTopicName"
    }
  },
  "Resources" : {
    "SNSTopic" : {
      "Type" : "AWS::SNS::Topic",
      "Properties" : {
        "TopicName" : { "Ref" : "TopicName"}
      }
```

```
    }
  }
}
```

Esempio 3: creare un argomento SNS inviando un file di parametri di esecuzione JSON con il tipo AMS ingest change

In questo esempio, inviate un file di parametri di esecuzione JSON con l'AMS ingest CT che crea l'argomento SNS. `TopicName` L'argomento SNS deve essere definito nel CloudFormation modello nel modo modificabile mostrato in questo esempio.

```
{
  "Name": "cfn-ingest",
  "Description": "CFNIngest Web Application Stack",
  "CloudFormationTemplateS3Endpoint": "$S3_PRESIGNED_URL",
  "VpcId": "VPC_ID",
  "Tags": [
    {"Key": "Enviroment Type", "Value": "Dev"}
  ],
  "Parameters": [
    {"Name": "TopicName", "Value": "MyTopic1"}
  ],
  "TimeoutInMinutes": 60
}
```

Esempio 4: invia un nuovo tipo di modifica che faccia riferimento allo stesso modello CloudFormation

Questo esempio JSON modifica il `TopicName` valore SNS senza apportare modifiche al modello. CloudFormation Al contrario, inviate un nuovo tipo di modifica Deployment | Ingestion | Stack from CloudFormation Template | Create change che faccia riferimento allo stesso modello CFN.

```
{
  "Name": "cfn-ingest",
  "Description": "CFNIngest Web Application Stack",
  "CloudFormationTemplateS3Endpoint": "$S3_PRESIGNED_URL",
  "VpcId": "VPC_ID",
  "Tags": [
    {"Key": "Enviroment Type", "Value": "Dev"}
  ],
  "Parameters": [
    {"Name": "TopicName", "Value": "MyTopic2"}
```

```
  ],
  "TimeoutInMinutes": 60
}
```

Esempio 5: utilizzate i valori dei parametri predefiniti nel modello CloudFormation

In questo esempio, SNS `TopicName` = 'MyTopicName' viene creato perché non è stato fornito alcun `TopicName` valore nel parametro di `Parameters` esecuzione. Se non fornite `Parameters` definizioni, vengono utilizzati i valori dei parametri predefiniti nel CloudFormation modello.

```
{
  "Name": "cfn-ingest",
  "Description": "CFNIngest Web Application Stack",
  "CloudFormationTemplateS3Endpoint": "$S3_PRESIGNED_URL",
  "VpcId": "VPC_ID",
  "Tags": [
    {"Key": "Enviroment Type", "Value": "Dev"}
  ],
  "TimeoutInMinutes": 60
}
```

CloudFormation Esempi di inserimento: applicazione Web a 3 livelli

Inserisci un CloudFormation modello per un'applicazione Web standard a 3 livelli.

Ciò include un Application Load Balancer, un gruppo target Application Load Balancer, un gruppo Auto Scaling, un modello di lancio del gruppo Auto Scaling, Amazon Relational Database Service (RDS for SQL Server) con un database MySQL, SSM Parameter Store e Secrets Manager. AWS AWS Attendi 30-60 minuti per esaminare questo esempio.

Prerequisiti

- Crea un segreto contenente un nome utente e una password con i valori corrispondenti utilizzando AWS Secrets Manager. Puoi fare riferimento a questo modello JSON di esempio (file zip) che contiene il nome `ams-shared/myapp/dev/dbsecrets` segreto e sostituirlo con il tuo nome segreto. Per informazioni sull'utilizzo di AWS Secrets Manager con AMS, vedereUtilizzo di AWS Secrets Manager con risorse AMS.
- Imposta i parametri richiesti in AWS SSM Parameter Store (PS). In questo esempio, le `VPCId` sottoreti private e pubbliche vengono archiviate nel PS SSM in percorsi come`/app/DemoApp/`

`PublicSubnet1a`,, `PublicSubnet1c` e. `Subnet-Id` `PrivateSubnet1a` `PrivateSubnet1c` `VPCCidr` Aggiorna i percorsi, i nomi e i valori dei parametri in base alle tue esigenze.

- Crea un ruolo di EC2 istanza Amazon IAM con autorizzazioni di lettura per i percorsi AWS Secrets Manager e SSM Parameter Store (il ruolo IAM creato e utilizzato in questi esempi è`customer-ec2_secrets_manager_instance_profile`). Se crei policy standard IAM come il ruolo del profilo dell'istanza, il nome del ruolo deve iniziare con. `customer-` Per creare un nuovo ruolo IAM, (puoi dargli un nome o un altro nome)`customer-ec2_secrets_manager_instance_profile`, usa il comando AMS change type Management | Applications | IAM instance profile | Create (ct-0ixp4ch2tiu04) CT e allega le policy richieste. Puoi esaminare le policy standard di AMS IAM `customer_secrets_manager_policy` e `customer_systemsmanager_parameterstore_policy` utilizzarle così come sono o come riferimento nella console AWS IAM.

Inserisci un CloudFormation modello per un'applicazione Web standard a 3 livelli

1. Carica il modello CloudFormation JSON di esempio allegato come file zip, tier-cfn-ingest3-.zip in un bucket S3 e genera un URL S3 firmato da utilizzare nella RFC di CFN Ingest. Per ulteriori informazioni, consulta presign. Il modello CFN può anche essere incluso copy/pasted nella RFC di CFN Ingest quando invii la RFC tramite la console AMS.
2. Crea un RFC di CloudFormation inserimento (Deployment | Ingestion | Stack from CloudFormation template | Create (ct-36cn2avfrrj9v)), tramite la console AMS o la CLI AMS. Il processo di automazione dell' CloudFormation inserimento convalida il modello per garantire che disponga di risorse valide supportate da AMS e rispetti gli standard di sicurezza. CloudFormation

   - Utilizzo della console: per il tipo di modifica, selezionate Deployment -> Ingestion -> Stack from CloudFormation Template -> Create, quindi aggiungete i seguenti parametri come esempio (tenete presente che l'impostazione predefinita per Multi è false): AZDatabase

```
CloudFormationTemplateS3Endpoint: "https://s3-ap-
southeast-2.amazonaws.com/amzn-s3-demo-bucket/3-tier-cfn-ingest.json?
AWSAccessKeyId=#{S3_ACCESS_KEY_ID}&Expires=#{EXPIRE_DATE}&Signature=#{SIGNATURE}"
VpcId: "VPC_ID"
TimeoutInMinutes: 120
IAMEC2InstanceProfile: "customer_ec2_secrets_manager_instance_profile"
MultiAZDatabase: "true"
WebServerCapacity: "2"
```

- Utilizzo di AWS CLI - Per informazioni dettagliate sulla creazione RFCs utilizzando il, consultate Creazione. AWS CLI RFCs Ad esempio, esegui il comando seguente:

```
aws --profile=saml amscm create-rfc  --change-type-id ct-36cn2avfrrj9v
 --change-type-version "2.0" --title "TEST_CFN_INGEST" --execution-
parameters "{\"CloudFormationTemplateS3Endpoint\":\"https://s3-
ap-southeast-2.amazonaws.com/my-bucket/3-tier-cfn-ingest.json?
AWSAccessKeyId=#{S3_ACCESS_KEY_ID}&Expires=#{EXPIRE_DATE}&Signature=#{SIGNATURE}\",
\"TimeoutInMinutes\":120,\"Description\":\"TEST\",\"VpcId\":\"VPC_ID\",
\"Name\":\"MY_TEST\",\"Tags\":[{\"Key\":\"env\",\"Value\":\"test\"}],
\"Parameters\":[{\"Name\":\"IAMEC2InstanceProfile\",\"Value\":
\"customer_ec2_secrets_manager_instance_profile\"},{\"Name\":\"MultiAZDatabase\",
\"Value\":\"true\"},{\"Name\":\"VpcId\",\"Value\":\"VPC_ID\"},{\"Name\":
\"WebServerCapacity\",\"Value\":\"2\"}]}" --endpoint-url https://amscm.us-
east-1.amazonaws.com/operational/ --no-verify-ssl
```

  Trova l'URL dell'Application Load Balancer nell'output di esecuzione AWS CloudFormation RFC per accedere al sito Web. Per informazioni sull'accesso alle risorse, consulta Accesso alle istanze.

Crea uno stack di importazione CloudFormation

Creazione di uno stack di importazione utilizzando la CloudFormation console

Per creare uno stack CloudFormation di importazione utilizzando la console

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina di RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina

Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.

- Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.

3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.

4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di uno CloudFormation stack di importazione utilizzando la CLI

Per creare uno stack CloudFormation di importazione utilizzando la CLI

1. Utilizza Inline Create (immetti un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

1. Prepara il CloudFormation modello che utilizzerai per creare lo stack e caricalo nel tuo bucket S3. Per dettagli importanti, consulta le linee guida, le CloudFormation best practice e le limitazioni di AWS Ingest.
2. Crea e invia la RFC ad AMS:

   - Crea e salva il file JSON dei parametri di esecuzione, includi i parametri del CloudFormation modello che desideri. L'esempio seguente lo chiama CreateCfnParams .json.

     Esempio di file stack CreateCfnParams di applicazioni Web .json:

```
{
  "Name": "cfn-ingest",
  "Description": "CFNIngest Web Application Stack",
  "VpcId": "VPC_ID",
  "CloudFormationTemplateS3Endpoint": "$S3_URL",
  "TimeoutInMinutes": 120,
  "Tags": [
   {
    "Key":   "Enviroment Type"
    "Value": "Dev",
   },
   {
    "Key":   "Application"
    "Value": "PCS",
   }
  ],
  "Parameters": [
   {
    "Name": "Parameter-for-S3Bucket-Name",
    "Value":  "BUCKET-NAME"
   },
```

```
    {
     "Name": "Parameter-for-Image-Id",
     "Value":  "AMI-ID"
    }
   ],
}
```

Esempio di file .json dell'argomento SNS: CreateCfnParams

```
{
  "Name": "cfn-ingest",
  "Description": "CFNIngest Web Application Stack",
  "CloudFormationTemplateS3Endpoint": "$S3_URL",
  "Tags": [
    {"Key": "Enviroment Type", "Value": "Dev"}
  ],
  "Parameters": [
    {"Name": "TopicName", "Value": "MyTopic1"}
  ]
}
```

3. Crea e salva il file JSON dei parametri RFC con il seguente contenuto. L'esempio seguente lo nomina CreateCfnRfc file.json:

```
{
   "ChangeTypeId": "ct-36cn2avfrrj9v",
   "ChangeTypeVersion": "2.0",
   "Title": "cfn-ingest"
}
```

4. Crea l'RFC, specificando il CreateCfnRfc file e il file: CreateCfnParams

```
aws amscm create-rfc --cli-input-json file://CreateCfnRfc.json  --execution-
parameters file://CreateCfnParams.json
```

Nella risposta ricevi l'ID della nuova RFC e puoi utilizzarlo per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

> **Note**
>
> Questo tipo di modifica è alla versione 2.0 ed è automatizzato (non eseguito manualmente). Ciò consente all'esecuzione CT di procedere più rapidamente e, un nuovo parametro CloudFormationTemplate, consente di incollare nella RFC un CloudFormation modello personalizzato. Inoltre, in questa versione, non alleghiamo i gruppi di sicurezza AMS predefiniti se si specificano i propri gruppi di sicurezza. Se non specificate i vostri gruppi di sicurezza nella richiesta, AMS allegherà i gruppi di sicurezza AMS predefiniti. In CFN Ingest v1.0, abbiamo sempre aggiunto i gruppi di sicurezza AMS predefiniti, indipendentemente dal fatto che abbiate fornito o meno i vostri gruppi di sicurezza.
> AMS ha abilitato 17 servizi AMS Self-Provisioned da utilizzare in questo tipo di modifica.
> Per informazioni sulle risorse supportate, consulta CloudFormation Ingest Stack: Supported Resources.

> **Note**
>
> La versione 2.0 accetta un endpoint S3 che non è un URL predefinito.
> Se utilizzi la versione precedente di questo CT, il valore del parametro CloudFormationTemplateS3Endpoint deve essere un URL predefinito.
> Comando di esempio per generare un URL del bucket S3 predefinito (Mac/Linux):

```
export S3_PRESIGNED_URL=$(aws s3 presign DASHDASHexpires-in 86400
 s3://BUCKET_NAME/CFN_TEMPLATE.json)
```

> Comando di esempio per generare un URL del bucket S3 predefinito (Windows):

```
for /f %i in ('aws s3 presign DASHDASHexpires-in 86400
 s3://BUCKET_NAME/CFN_TEMPLATE.json') do set S3_PRESIGNED_URL=%i
```

> Vedi anche Creazione di bucket prefirmati URLs per Amazon S3.

> **Note**
>
> Se il bucket S3 esiste in un account AMS, devi usare le tue credenziali AMS per questo comando. Ad esempio, potrebbe essere necessario aggiungere `--profile saml` dopo aver ottenuto le credenziali AMS AWS Security Token Service ().AWS STS

Tipi di modifiche correlati:, CloudFormation Approva un set di modifiche allo stack di importazione Aggiorna lo stack di importazione AWS CloudFormation

Per ulteriori informazioni su AWS CloudFormation, consulta AWS CloudFormation. Per visualizzare i CloudFormation modelli, apri AWS CloudFormation Template Reference.

Convalida di un inserimento AWS CloudFormation

Il modello è convalidato per garantire che possa essere creato in un account AMS. Se supera la convalida, viene aggiornato per includere tutte le risorse o le configurazioni necessarie per renderlo conforme ad AMS. Ciò include l'aggiunta di risorse come gli CloudWatch allarmi Amazon per consentire ad AMS Operations di monitorare lo stack.

La RFC viene rifiutata se si verifica una delle seguenti condizioni:

- La sintassi RFC JSON non è corretta o non segue il formato specificato.
- L'URL prefirmato del bucket S3 fornito non è valido.
- Il modello non è una sintassi valida. AWS CloudFormation
- Nel modello non sono impostati valori predefiniti per tutti i valori dei parametri.
- Il modello non supera la convalida AMS. Per le fasi di convalida AMS, consulta le informazioni più avanti in questo argomento.

La RFC fallisce se lo CloudFormation stack non viene creato a causa di un problema di creazione delle risorse.

Per saperne di più sulla convalida e la validazione CFN, consulta Template Validation and CloudFormation ingest stack: esempi di validatori CFN.

Aggiorna lo stack di importazione AWS CloudFormation

Aggiornamento di uno stack di importazione tramite CloudFormation la console

Per aggiornare uno stack CloudFormation di importazione utilizzando la console

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.
4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Aggiornamento di uno stack CloudFormation di importazione tramite la CLI

Per aggiornare uno stack CloudFormation di importazione utilizzando la CLI

1. Utilizza Inline Create (immetti un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di

esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.

2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

1. Prepara il AWS CloudFormation modello che desideri utilizzare per aggiornare lo stack e caricalo nel tuo bucket S3. Per dettagli importanti, consulta le linee guida, le CloudFormation best practice e le limitazioni di AWS Ingest.
2. Crea e invia la RFC ad AMS:

   - Crea e salva il file JSON dei parametri di esecuzione, includi i parametri del CloudFormation modello che desideri. Questo esempio lo chiama UpdateCfnParams .json.

     Esempio di UpdateCfnParams file.json con aggiornamenti dei parametri in linea:

     ```
     {
       "StackId": "stack-yjjoo9aicjyqw4ro2",
       "VpcId": "VPC_ID",
       "CloudFormationTemplate": "{\"AWSTemplateFormatVersion\":\"2010-09-09\",
     \"Description\":\"Create a SNS topic\",\"Parameters\":{\"TopicName\":{\"Type
     \":\"String\"},\"DisplayName\":{\"Type\":\"String\"}},\"Resources\":{\"SnsTopic
     \":{\"Type\":\"AWS::SNS::Topic\",\"Properties\":{\"TopicName\":{\"Ref\":
     \"TopicName\"},\"DisplayName\":{\"Ref\":\"DisplayName\"}}}}}",
     ```

```
  "TemplateParameters": [
    {
      "Key": "TopicName",
      "Value": "TopicNameCLI"
    },
    {
      "Key": "DisplayName",
      "Value": "DisplayNameCLI"
    }
  ],
  "TimeoutInMinutes": 1440
}
```

Esempio di UpdateCfnParams file.json con endpoint bucket S3 contenente un modello aggiornato: CloudFormation

```
{
  "StackId": "stack-yjjoo9aicjyqw4ro2",
  "VpcId": "VPC_ID",
  "CloudFormationTemplateS3Endpoint": "s3_url",
  "TemplateParameters": [
    {
      "Key": "TopicName",
      "Value": "TopicNameCLI"
    },
    {
      "Key": "DisplayName",
      "Value": "DisplayNameCLI"
    }
  ],
  "TimeoutInMinutes": 1080
}
```

3. Crea e salva il file JSON dei parametri RFC con il seguente contenuto. Questo esempio lo chiama UpdateCfnRfc file.json.

```
{
   "ChangeTypeId": "ct-361tlo1k7339x",
   "ChangeTypeVersion": "1.0",
   "Title": "cfn-ingest-template-update"
}
```

4. Crea la RFC, specificando il UpdateCfnRfc file e il file: UpdateCfnParams

```
aws amscm create-rfc --cli-input-json file://UpdateCfnRfc.json  --execution-
parameters file://UpdateCfnParams.json
```

Nella risposta ricevi l'ID della nuova RFC e puoi utilizzarlo per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

- Questo tipo di modifica è ora alla versione 2.0. Le modifiche includono la rimozione del AutoApproveUpdateForResourcesparametro, utilizzato nella versione 1.0 di questo CT, e l'aggiunta di due nuovi parametri: AutoApproveRiskyUpdatese BypassDriftCheck.
- Se il bucket S3 esiste in un account AMS, è necessario utilizzare le credenziali AMS per questo comando. Ad esempio, potrebbe essere necessario aggiungere `--profile saml` dopo aver ottenuto le credenziali AMS AWS Security Token Service ().AWS STS
- Tutti `Parameter` i valori per le risorse nel CloudFormation modello devono avere un valore, tramite un valore predefinito o personalizzato tramite la sezione dei parametri del CT. È possibile sovrascrivere il valore del parametro strutturando le risorse del CloudFormation modello in modo che facciano riferimento a una chiave Parameters. Per esempi che mostrano come fare, consulta CloudFormation ingest stack: esempi di validatori CFN.

  IMPORTANTE: i parametri mancanti non vengono forniti esplicitamente nel modulo, per impostazione predefinita sono i valori attualmente impostati nello stack o nel modello esistente.
- Per un elenco dei servizi forniti autonomamente che è possibile aggiungere utilizzando AWS CloudFormation Ingest, consulta CloudFormation Ingest Stack: Supported Resources.

  Per ulteriori informazioni AWS CloudFormation, consulta AWS CloudFormation.

Convalida di un inserimento AWS CloudFormation

Il modello è convalidato per garantire che possa essere creato in un account AMS. Se supera la convalida, viene aggiornato per includere tutte le risorse o le configurazioni necessarie per renderlo conforme ad AMS. Ciò include l'aggiunta di risorse come gli CloudWatch allarmi Amazon per consentire ad AMS Operations di monitorare lo stack.

La RFC viene rifiutata se si verifica una delle seguenti condizioni:

- La sintassi RFC JSON non è corretta o non segue il formato specificato.
- L'URL prefirmato del bucket S3 fornito non è valido.
- Il modello non è una sintassi valida. AWS CloudFormation
- Nel modello non sono impostati valori predefiniti per tutti i valori dei parametri.
- Il modello non supera la convalida AMS. Per le fasi di convalida AMS, consulta le informazioni più avanti in questo argomento.

La RFC fallisce se lo CloudFormation stack non viene creato a causa di un problema di creazione delle risorse.

Per saperne di più sulla convalida e la validazione CFN, consulta Template Validation and CloudFormation ingest stack: esempi di validatori CFN.

CloudFormation Approva un set di modifiche allo stack di importazione

Approvazione e aggiornamento di uno stack di importazione tramite CloudFormation la console

Per approvare e aggiornare uno stack di importazione CloudFormation utilizzando la console

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina di RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella

vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.

4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Approvazione e aggiornamento di uno stack CloudFormation di importazione tramite la CLI

Per approvare e aggiornare uno stack di CloudFormation importazione utilizzando la CLI

1. Utilizzate Inline Create (emettete un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (create due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed emettete il comando con i due file come input. `create-rfc` Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla

parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

1. Invia lo schema JSON dei parametri di esecuzione per questo tipo di modifica in un file nella cartella corrente. Questo esempio lo chiama CreateAsgParams .json:

```
aws amscm create-rfc --change-type-id "ct-1404e21baa2ox" --change-
type-version "1.0" --title "Approve Update" --execution-parameters
 file://PATH_TO_EXECUTION_PARAMETERS --profile saml
```

2. Modificate e salvate lo schema come segue:

```
{
  "StackId": "STACK_ID",
  "VpcId": "VPC_ID",
  "ChangeSetName": "UPDATE-ef81e2bc-03f6-4b17-a3c7-feb700e78faa",
  "TimeoutInMinutes": 1080
}
```

Suggerimenti

> **Note**
>
> Se ci sono più risorse in uno stack e desideri eliminare solo un sottoinsieme delle risorse dello stack, usa CloudFormation Update CT; vedi CloudFormation Ingest Stack: Update. Puoi anche inviare un caso di richiesta di assistenza e gli ingegneri AMS possono aiutarti a creare il changeset, se necessario.

Per saperne di più AWS CloudFormation, consulta. AWS CloudFormation

Update AWS CloudFormation stack: protezione dalla terminazione

Aggiornamento di uno stack di protezione dalle AWS CloudFormation terminazioni con la console

Di seguito viene illustrato questo tipo di modifica nella console AMS.

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.
4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Aggiornamento di una protezione dalla terminazione AWS CloudFormation dello stack con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di

esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.

2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

Specificate solo i parametri che desiderate modificare. I parametri assenti mantengono i valori esistenti.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws amscm create-rfc \
--change-type-id "ct-2uzbqr7x7mekd" \
--change-type-version "1.0" \
--title "Enable termination protection on CFN stack" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
ManageResourceTerminationProtection\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ResourceId\":[\"stack-psvnq6cupymio3enl\"],\"TerminationProtectionDesiredState\":
[\"enabled\"]}}"
```

CREAZIONE DEL MODELLO:

1. Esporta i parametri di esecuzione per questo tipo di modifica in un file JSON; questo esempio lo chiama EnableTermPro CFNParams .json:

```
aws amscm get-change-type-version --change-type-id "ct-2uzbqr7x7mekd"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 EnableTermProCFNParams.json
```

2. Modificate e salvate il EnableTermPro CFNParams file, mantenendo solo i parametri che desiderate modificare. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
  "DocumentName": "AWSManagedServices-ManageResourceTerminationProtection",
  "Region": "us-east-1",
  "Parameters": {
    "ResourceId": ["stack-psvnq6cupymio3enl"],
    "TerminationProtectionDesiredState": ["enabled"]
  }
}
```

3. Esporta il modello RFC in un file nella cartella corrente; questo esempio lo chiama EnableTermPro CFNRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > EnableTermProCFNRfc.json
```

4. Modifica e salva il file.json. EnableTermPro CFNRfc Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
    "ChangeTypeId": "ct-2uzbqr7x7mekd",
    "ChangeTypeVersion": "1.0",
    "Title": "Enable termination protection on CFN instance"
}
```

5. Crea la RFC, specificando il EnableTermPro CFNRfc file e il EnableTermPro CFNParams file:

```
aws amscm create-rfc --cli-input-json file://EnableTermProCFNRfc.json  --execution-
parameters file://EnableTermProCFNParams.json
```

Nella risposta ricevi l'ID della nuova RFC e puoi utilizzarlo per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

> **Note**
>
> Esiste un CT correlato per Amazon EC2, EC2 stack: Updating termination protection.

Per ulteriori informazioni sulla protezione dalla terminazione, consulta Proteggere uno stack dall'eliminazione.

Implementazioni IAM automatizzate che utilizzano CFN ingest o stack update in AMS CTs

Puoi utilizzare questi tipi di modifica AMS per implementare i ruoli IAM (la `AWS::IAM::Role` risorsa) sia nella landing zone multi-account (MALZ) che nella landing zone a account singolo (SALZ):

- Distribuzione | Ingestione | Stack from Template | Crea (ct-36cn2avfrrj9v) CloudFormation
- Gestione | Stack personalizzato | Stack From Template | Aggiornamento (ct-361tlo1k7339x) CloudFormation
- Gestione | Custom Stack | Stack From Template | Approva e aggiorna (ct-1404e21baa2ox) CloudFormation

Convalide eseguite sui ruoli IAM nel tuo modello CFN:

- ManagedPolicyArns: L'attributo non ManagedPolicyArnsdeve esistere in. `AWS::IAM::Role` La convalida non consente di allegare politiche gestite al ruolo assegnato. Invece, le autorizzazioni per il ruolo possono essere gestite utilizzando la politica in linea tramite la proprietà Policies.
- PermissionsBoundary: La politica utilizzata per impostare il limite delle autorizzazioni per il ruolo può essere solo la politica gestita fornita da AMS:. `AWSManagedServices_IAM_PermissionsBoundary` Questa politica funge da guard rail che protegge le risorse dell'infrastruttura AMS dalla modifica in base al ruolo assegnato. Con questo limite di autorizzazioni predefinito, i vantaggi in termini di sicurezza offerti da AMS vengono preservati.

`AWSManagedServices_IAM_PermissionsBoundary`(impostazione predefinita) è obbligatoria, in caso contrario la richiesta viene rifiutata.

- MaxSessionDuration: La durata massima della sessione che può essere impostata per il ruolo IAM è compresa tra 1 e 4 ore. Lo standard tecnico AMS richiede l'accettazione del rischio da parte del cliente per una durata della sessione superiore a 4 ore.
- RoleName: I seguenti namespace sono conservati da AMS e non possono essere utilizzati come prefissi dei nomi dei ruoli IAM:

```
AmazonSSMRole,
AMS,
Ams,
ams,
AWSManagedServices,
customer_developer_role,
customer-mc-,
Managed_Services,
MC,
Mc,
mc,
SENTINEL,
Sentinel,
sentinel,
StackSet-AMS,
StackSet-Ams,
StackSet-ams,
StackSet-AWS,
StackSet-MC,
StackSet-Mc,
StackSet-mc
```

- Politiche: la policy in linea incorporata nel ruolo IAM può includere solo una serie di azioni IAM preapprovate da AMS. Questo è il limite superiore di tutte le azioni IAM a cui è consentito creare un ruolo IAM con (policy di controllo). La politica di controllo è composta da:
  - Tutte le azioni incluse nella policy AWS gestita ReadOnlyAccess che fornisce accesso in sola lettura a tutte le risorse Servizi AWS
  - Le seguenti azioni, con la limitazione delle azioni S3 tra account, ad esempio le azioni S3 consentite, possono essere eseguite solo sulle risorse presenti nello stesso account del ruolo creato:

```
amscm:*,
amsskms:*,
lambda:InvokeFunction,
logs:CreateLogStream,
logs:PutLogEvents,
s3:AbortMultipartUpload,
s3:DeleteObject,
s3:DeleteObjectVersion,
s3:ObjectOwnerOverrideToBucketOwner,
s3:PutObject,
s3:ReplicateTags,
secretsmanager:GetRandomPassword,
sns:Publish
```

Qualsiasi ruolo IAM creato o aggiornato tramite CFN ingest può consentire azioni elencate in questa politica di controllo o azioni che rientrano nell'ambito (meno permissivo) delle azioni elencate nella politica di controllo. Attualmente consentiamo queste azioni IAM sicure che possono essere classificate come azioni di sola lettura, oltre alle azioni non di sola lettura sopra menzionate che non possono essere eseguite e sono preapprovate in base allo standard tecnico AMS. CTs

- AssumeRolePolicyDocument: Le seguenti entità sono preapprovate e possono essere incluse nella politica di fiducia per assumere il ruolo che viene creato:
  - Qualsiasi entità IAM (ruolo, utente, utente root, sessione con ruolo assunto da STS) nello stesso account può assumere il ruolo.
  - Il ruolo Servizi AWS può essere assunto da:

```
apigateway.amazonaws.com,
autoscaling.amazonaws.com,
cloudformation.amazonaws.com,
codebuild.amazonaws.com,
codedeploy.amazonaws.com,
codepipeline.amazonaws.com,
datapipeline.amazonaws.com,
datasync.amazonaws.com,
dax.amazonaws.com,
dms.amazonaws.com,
ec2.amazonaws.com,
ecs-tasks.amazonaws.com,
ecs.application-autoscaling.amazonaws.com,
```

```
elasticmapreduce.amazonaws.com,
es.amazonaws.com,
events.amazonaws.com,
firehose.amazonaws.com,
glue.amazonaws.com,
lambda.amazonaws.com,
monitoring.rds.amazonaws.com,
pinpoint.amazonaws.com,
rds.amazonaws.com,
redshift.amazonaws.com,
s3.amazonaws.com,
sagemaker.amazonaws.com,
servicecatalog.amazonaws.com,
sns.amazonaws.com,
ssm.amazonaws.com,
states.amazonaws.com,
storagegateway.amazonaws.com,
transfer.amazonaws.com,
vmie.amazonaws.com
```

- Il provider SAML dello stesso account può assumere il ruolo. Attualmente, l'unico nome di provider SAML supportato è. `customer-saml`

Se una o più convalide falliscono, la RFC viene rifiutata. Un esempio di motivo di rifiuto RFC è il seguente:

```
{"errorMessage":"[ 'LambdaRole: The maximum session duration (in seconds) should be a
 numeric value in the range 3600 to 14400 (i.e. 1 to 4 hours).', 'lambda-policy: Policy
 document is too permissive.']","errorType":"ClientError"}
```

Se hai bisogno di assistenza per una convalida o un'esecuzione RFC non riuscita, utilizza la corrispondenza RFC per contattare AMS. Per istruzioni, consulta Corrispondenza e allegato RFC (console). Per qualsiasi altra domanda, invia una richiesta di assistenza. Per istruzioni, consulta Creazione di una richiesta di assistenza.

> **Note**
>
> Al momento non applichiamo alcuna best practice IAM nell'ambito delle nostre convalide IAM. Per le migliori pratiche IAM, consulta Best practice di sicurezza in IAM.

Creazione di ruoli IAM con azioni più permissive o applicazione delle best practice IAM

Crea le tue entità IAM con i seguenti tipi di modifiche manuali:

- Distribuzione | Componenti stack avanzati | Identity and Access Management (IAM) | Creare entità o policy (ct-3dpd8mdd9jn1r)
- Gestione | Componenti dello stack avanzati | Identity and Access Management (IAM) | Aggiornamento di entità o policy (ct-27tuth19k52b4)

Ti consigliamo di leggere e comprendere i nostri standard tecnici prima di archiviare questi manuali. RFCs Per l'accesso, vedi Come accedere agli standard tecnici.

> **Note**
>
> Ogni ruolo IAM creato direttamente con questi tipi di modifiche manuali appartiene al proprio stack individuale e non risiede nello stesso stack in cui vengono create le altre risorse dell'infrastruttura tramite CFN Ingest CT.

Aggiornamento dei ruoli IAM creati con CFN ingest tramite tipi di modifica manuali quando gli aggiornamenti non possono essere eseguiti tramite tipi di modifica automatici

Utilizza Management | Advanced stack components | Identity and Access Management (IAM) | Aggiorna il tipo di modifica dell'entità o della policy (ct-27tuth19k52b4).

> **Important**
>
> Gli aggiornamenti sui ruoli IAM tramite il CT manuale non si riflettono nei modelli dello stack CFN e causano una deriva dello stack. Una volta che il ruolo è stato aggiornato tramite una richiesta manuale a uno stato che non supera le nostre convalide, il ruolo non può essere ulteriormente aggiornato utilizzando nuovamente lo Stack Update CT (ct-361tlo1k7339x) purché continui a non essere conforme alle nostre convalide. L'aggiornamento CT può essere utilizzato solo se il modello dello stack CFN è conforme alle nostre convalide. Tuttavia, lo stack può ancora essere aggiornato tramite Stack Update CT (ct-361tlo1k7339x), a condizione che la risorsa IAM non conforme alle nostre convalide non venga aggiornata e il modello CFN superi le nostre convalide.

Eliminazione dei ruoli IAM creati tramite ingest AWS CloudFormation

Se desideri eliminare l'intero stack, utilizza il seguente tipo di modifica automatica di Delete Stack. Per istruzioni, consulta Delete Stack:

- ID del tipo di modifica: ct-0q0bic0ywqk6c
- Classificazione: Gestione | Stack standard | Stack | Eliminazione e gestione | Componenti avanzati dello stack | Stack | Elimina

Se desideri eliminare un ruolo IAM senza eliminare l'intero stack, puoi rimuovere il ruolo IAM dal CloudFormation modello e utilizzare il modello aggiornato come input per il tipo di modifica automatizzato Stack Update:

- ID del tipo di modifica: ct-361tlo1k7339x
- Classificazione: Gestione | Stack personalizzato | Stack da modello | Aggiornamento CloudFormation

Per istruzioni, consulta Update AWS CloudFormation ingest stack.

CodeDeploy richieste

Puoi usare AWS CodeDeploy per creare contenitori di applicazioni che puoi poi distribuire tramite un gruppo di CodeDeploy applicazioni. Per ulteriori informazioni CodeDeploy, consulta AWS CodeDeploy Documentation.

Lavorare con AWS CodeDeploy prevede il seguente processo:

1. Crea un' CodeDeploy applicazione. L' CodeDeploy applicazione è un nome o contenitore utilizzato da CodeDeploy per garantire che durante una distribuzione venga fatto riferimento alla revisione, alla configurazione di distribuzione e al gruppo di distribuzione corretti.
2. Crea un gruppo di CodeDeploy distribuzione. Un gruppo CodeDeploy di distribuzione definisce un insieme di istanze individuali destinate a una distribuzione. AMS ha un tipo di modifica separato per i gruppi CodeDeploy di distribuzione per EC2.
3. Distribuisci l' CodeDeploy applicazione tramite il gruppo CodeDeploy di distribuzione.

CodeDeploy applicazione

Crea o distribuisci applicazioni. CodeDeploy

Creare un'applicazione CodeDeploy

Creazione di un' CodeDeploy applicazione con la console

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.
4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.

5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di un' CodeDeploy applicazione con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws amscm create-rfc --change-type-id "ct-0ah3gwb9seqk2" --change-type-version "1.0"
 --title "Stack-Create-CD-App" --execution-parameters "{\"Description\":\"TestCdApp\",
\"VpcId\":\"VPC_ID\",\"StackTemplateId\":\"stm-sft6rv00000000000\",\"Name\":\"Test\",
\"TimeoutInMinutes\":60,\"Parameters\":{\"CodeDeployApplicationName\":\"Test\"}}"
```

CREAZIONE DEL MODELLO:

1. Esporta lo schema JSON dei parametri di esecuzione per l' CodeDeploy applicazione CT in un file nella cartella corrente; questo esempio lo chiama Create CDApp Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0ah3gwb9seqk2" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateCDAppParams.json
   ```

2. Modificate e salvate il file JSON come segue. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

   ```
   {
   "Description":                      "Create WP CodeDeploy App",
   "VpcId":                            "VPC_ID",
   "StackTemplateId":                  "stm-sft6rv00000000000",
   "Name":                             "WpCDApp",
   "TimeoutInMinutes":                 60,
   "Parameters":   {
       "CodeDeployApplicationName":    "WordPressCDApp"
       }
   }
   ```

3. Esporta il modello JSON CreateRfc per in un file nella cartella corrente; questo esempio lo chiama Create CDApp RFC.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateCDAppRfc.json
   ```

4. Modificate e salvate il file JSON come segue. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0ah3gwb9seqk2",
   "Title":                "CD-App-Stack-RFC"
   }
   ```

5. Crea la RFC, specificando il file Create CDApp Rfc e il file dei parametri di esecuzione:

```
aws amscm create-rfc --cli-input-json file://CreateCDAppRfc.json --execution-
parameters file://CreateCDAppParams.json
```

   Nella risposta ricevi l'ID della nuova RFC e puoi utilizzarlo per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

Per ulteriori informazioni su AWS CodeDeploy, consulta Create an Application with AWS CodeDeploy.

Distribuisci l'applicazione CodeDeploy

Distribuzione di un' CodeDeploy applicazione con la console

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella

vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.

4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Distribuzione di un' CodeDeploy applicazione con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla

parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws amscm create-rfc --change-type-id "ct-2edc3sd1sqmrb" --change-
type-version "2.0" --title "Stack-Deploy-CD-App" --execution-
parameters "{\"Description\":\"MyCDAppDeployTest\",\"VpcId\":
\"VPC_ID\",\"Name\":\"Test\",\"TimeoutInMinutes\":60,\"Parameters\":
{\"CodeDeployApplicationName\":\"TestCDApp\",\"CodeDeployDeploymentConfigName\":
\"CodeDeployDefault.OneAtATime\",\"CodeDeployDeploymentGroupName\":\"TestCDDepGroup\",
\"CodeDeployIgnoreApplicationStopFailures\":false,\"CodeDeployRevision\":
{\"RevisionType\":\"S3\",\"S3Location\":{\"S3Bucket\":\"amzn-s3-demo-bucket\",
\"S3BundleType\":\"tar\",\"S3Key\":\"TestKey\"}}}}"Test\"}}"
```

CREAZIONE DEL MODELLO:

1. Visualizza lo schema JSON dei parametri di esecuzione per la distribuzione dell' CodeDeploy applicazione CT; questo esempio lo chiama Deploy CDApp Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2edc3sd1sqmrb" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeployCDAppParams.json
   ```

2. Modificate il file JSON come segue. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

   ```
   {
   "Description":                    "Deploy WordPress CodeDeploy Application",
   "VpcId":                          "VPC_ID",
   "Name":                           "WP CodeDeploy Deployment Group",
   "TimeoutInMinutes":               360,
   "Parameters":   {
       "CodeDeployApplicationName":        "WordPressCDApp",
       "CodeDeployDeploymentGroupName":    "WordPressCDDepGroup",
       "CodeDeployIgnoreApplicationStopFailures": false,
       "CodeDeployRevision": {
   ```

```
      "RevisionType": "S3",
      "S3Location": {
        "S3Bucket": "amzn-s3-demo-bucket",
        "S3BundleType": "zip",
        "S3Key": "wordpress.zip" }
        }
    }
}
```

3. Esporta il modello JSON CreateRfc per in un file nella cartella corrente; questo esempio lo chiama Deploy CDApp RFC.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeployCDAppRfc.json
```

4. Modifica e salva il file Deploy RFC.json. CDApp Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-2edc3sd1sqmrb",
"Title":                "CD-Deploy-For-CD-APP-Stack-RFC"
}
```

5. Crea la RFC, specificando il file dei parametri di esecuzione e il file Deploy CDApp Rfc:

```
aws amscm create-rfc --cli-input-json file://DeployCDAppRfc.json  --execution-
parameters file://DeployCDAppParams.json
```

   Nella risposta si riceve l'ID della nuova RFC e si può utilizzare per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

Per ulteriori informazioni, consulta Creare una distribuzione con CodeDeploy.

CodeDeploy gruppi di distribuzione

Creare gruppi di CodeDeploy applicazioni.

Creare un gruppo CodeDeploy di distribuzione

Creazione di un gruppo di CodeDeploy distribuzione con la console

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un Oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.
4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di un gruppo di CodeDeploy distribuzione con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws amscm create-rfc --change-type-id "ct-2gd0u847qd9d2" --change-type-version
 "1.0" --title "Stack-Create-CD-Dep-Group" --execution-parameters "{\"Description
\":\"TestCdDepGroupRfc\",\"VpcId\":\"VPC_ID\",\"StackTemplateId\":\"stm-
sp9lrk00000000000\",\"Name\":\"MyTestCDDepGroup\",\"TimeoutInMinutes\":60,\"Parameters
\":{\"CodeDeployApplicationName\":\"TestCDApp\",\"CodeDeployAutoScalingGroups\":
[\"TestASG\"],\"CodeDeployDeploymentConfigName\":\"CodeDeployDefault.OneAtATime\",
```

```
\"CodeDeployDeploymentGroupName\":\"Test\",\"CodeDeployServiceRoleArn\":
\"arn:aws:iam::000000000:role/aws-codedeploy-role\"}}"
```

CREAZIONE DEL MODELLO:

1. Invia lo schema JSON dei parametri di esecuzione in un file nella cartella corrente; questo esempio lo chiama Create CDDep GroupParams .json:

```
aws amscm get-change-type-version --change-type-id "ct-2gd0u847qd9d2"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateCDDepGroupParams.json
```

2. Modifica e salva il file JSON. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"Description":                      "CreateCDDeploymentGroup",
"VpcId":                            "VPC_ID",
"StackTemplateId":                  "stm-sp9lrk00000000000",
"Name":                             "WordPressCDAppGroup",
"TimeoutInMinutes":                 60,
"Parameters":   {
    "CodeDeployApplicationName":        "WordPressCDApp",
    "CodeDeployAutoScalingGroups":      ["ASG_NAME"],
    "CodeDeployDeploymentConfigName":   "CodeDeployDefault.HalfAtATime",
    "CodeDeployDeploymentGroupName":    "UNIQUE_CDDepGroupNAME",
    "CodeDeployServiceRoleArn":         "arn:aws:iam::ACCOUNT_ID:role/aws-
codedeploy-role"
    }
}
```

3. Esporta il modello JSON CreateRfc per in un file nella cartella corrente; questo esempio lo chiama Create CDDep GroupRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateCDDepGroupRfc.json
```

4. Modifica e salva il file JSON. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2gd0u847qd9d2",
"Title":                "CD-Dep-Group-RFC"
}
```

5. Create la RFC, specificando il CDDep GroupRfc file Create e il file dei parametri di esecuzione:

```
aws amscm create-rfc --cli-input-json file://CreateCDDepGroupRfc.json --execution-
parameters file://CreateCDDepGroupParams.json
```

   Nella risposta ricevi l'ID della nuova RFC e puoi utilizzarlo per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

Per ulteriori informazioni sui gruppi di CodeDeploy distribuzione AWS, consulta Create a Deployment Group with AWS CodeDeploy.

Crea un gruppo di CodeDeploy distribuzione per EC2

Creazione di un gruppo CodeDeploy di distribuzione per EC2 con la console

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un Oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella

vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.

4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di un gruppo di CodeDeploy distribuzione per EC2 con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla

parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws amscm create-rfc --change-type-id "ct-00tlkda4242x7" --change-type-
version "1.0" --title "Stack-Create-CD-Ec2-Dep-Group" --execution-parameters
 "{\"Description\":\"MyTestCdDepEc2DepGroup\",\"VpcId\":\"VPC_ID\",\"Name\":
\"TestCDDepEc2Group\",\"StackTemplateId\":\"stm-n3hsoirgqeqqdbpk2\",\"TimeoutInMinutes
\":60,\"Parameters\":{\"ApplicationName\":\"TestCDApp\",\"DeploymentConfigName\":
\"CodeDeployDefault.OneAtATime\",\"AutoRollbackEnabled\":\"False\",\"EC2FilterTag\":
\"Name=Test\",\"EC2FilterTag2\":\"\",\"EC2FilterTag3\":\"\",\"ServiceRoleArn\":\"\"}}"
```

CREAZIONE DEL MODELLO:

1. Invia lo schema JSON dei parametri di esecuzione in un file; questo esempio lo chiama Create CDDep GroupEc 2Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-00tlkda4242x7"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateCDDepGroupEc2Params.json
   ```

2. Modifica e salva il file JSON. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

   ```
   {
   "Description":                    "CreateCDDepGroupEc2",
   "VpcId":                          "VPC_ID",
   "StackTemplateId":                "stm-n3hsoirgqeqqdbpk2",
   "Name":                           "CDAppGroupEc2",
   "TimeoutInMinutes":               60,
   "Parameters":   {
       "ApplicationName":        "CDAppEc2",
       "DeploymentConfigName":   "CodeDeployDefault.OneAtATime",
       "CodeDeployDeploymentGroupName":    "UNIQUE_CDDepGroupNAME",
       "CodeDeployServiceRoleArn":         "arn:aws:iam::ACCOUNT_ID:role/aws-
   codedeploy-role"
       }
   ```

```
}
```

3. Esporta il modello JSON CreateRfc per in un file nella cartella corrente; questo esempio lo chiama Create CDDep GroupEc 2RFC.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateCDDepGroupEc2Rfc.json
```

4. Modifica e salva il file JSON. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-00tlkda4242x7",
"Title":                "CD-Dep-Group-For-Ec2-Stack-RFC"
}
```

5. Crea la RFC, specificando il file Create CDDep GroupEc 2Rfc e il file dei parametri di esecuzione:

```
aws amscm create-rfc --cli-input-json file://CreateCDDepGroupEc2Rfc.json --
execution-parameters file://CreateCDDepGroupEc2Params.json
```

   Nella risposta ricevi l'ID della nuova RFC e puoi utilizzarlo per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

Per ulteriori informazioni sui gruppi di CodeDeploy distribuzione AWS, consulta Create a Deployment Group with AWS CodeDeploy.

AWS Database Migration Service (AWS DMS)

AWS Database Migration Service (AWS DMS) consente di migrare i database verso AMS in modo semplice e sicuro. È possibile eseguire la migrazione dei dati da e verso la maggior parte dei database commerciali e open source più diffusi, tra cui Oracle, MySQL e PostgreSQL. Il servizio supporta migrazioni omogenee come da Oracle a Oracle e anche migrazioni eterogenee tra diverse piattaforme di database, come Oracle a PostgreSQL o MySQL a Oracle. AWS DMS è un AWS servizio; l'AMS ti aiuta a creare risorse nel tuo account gestito da AMS CTs AWS DMS

L'immagine seguente mostra il flusso di lavoro di una migrazione di database.

Argomenti

AWS Database Migration Service (AWS DMS), prima di iniziare

Quando pianificate una migrazione di database utilizzando AMS AWS DMS, tenete presente quanto segue:

- Endpoint di origine e destinazione: è necessario sapere quali informazioni e tabelle del database di origine devono essere migrate nel database di destinazione. AMS AWS DMS supporta la migrazione di base dello schema, inclusa la creazione di tabelle e chiavi primarie. Tuttavia, AMS AWS DMS non crea automaticamente indici secondari, chiavi esterne, account e così via nel database di destinazione. Per ulteriori informazioni, consulta Sorgenti per la migrazione dei dati e Target per la migrazione dei dati.
- Migrazione schema/codice: AMS AWS DMS non esegue conversioni di schemi o codici. Puoi utilizzare strumenti quali Oracle SQL Developer, MySQL Workbench o pgAdmin III per convertire lo schema. Se desideri convertire uno schema esistente in un motore di database diverso, puoi utilizzare AWS Schema Conversion Tool. Questo può creare uno schema di destinazione e, inoltre, generare e creare un intero schema: tabelle, indici, viste e così via. Puoi anche usare lo strumento per convertire PL/SQL TSQL in pgSQL e altri formati.
- Tipi di dati non supportati: alcuni tipi di dati di origine devono essere convertiti nei tipi di dati equivalenti per il database di destinazione.

AWS DMS scenari da considerare

I seguenti scenari, documentati, potrebbero aiutarti a creare il tuo percorso di migrazione del database.

- Migrazione dei dati da un server MySQL locale ad Amazon RDS MySQL: consulta il post sul blog AWS Migrate On-Premises MySQL Data to Amazon RDS (e viceversa)
- Migrazione dei dati da un database Oracle al database Amazon RDS Aurora PostgreSQL: consulta il post sul blog di AWS Una rapida introduzione alla migrazione da un database Oracle a un database Amazon Aurora PostgreSQL

- Migrazione dei dati da RDS MySQL a S3: consulta il post sul blog di AWS Come archiviare i dati dai database relazionali su Amazon Glacier usando AWS DMS

Per la migrazione di un database, è necessario fare quanto segue:

- Pianifica la migrazione del database, inclusa la configurazione di un sottogruppo di replica.
- Alloca un'istanza di replica che esegua tutti i processi per la migrazione.
- Specificare un endpoint del database di origine e uno di destinazione.
- Crea un'attività o una serie di attività per definire le tabelle e i processi di replica da utilizzare.
- Crea l' AWS DMS IAM `dms-cloudwatch-logs-role` e `dms-vpc-role` i ruoli. Se utilizzi Amazon Redshift come database di destinazione, devi anche creare e aggiungere il ruolo IAM `dms-access-for-endpoint` al tuo account AWS. Per ulteriori informazioni, consulta Creazione dei ruoli IAM da utilizzare con l'AWS CLI e l'API AWS DMS.

Queste procedure dettagliate forniscono un esempio di utilizzo della console AMS o della CLI AMS per creare un (). AWS Database Migration Service AWS DMS Vengono forniti i comandi CLI per creare l'istanza di AWS DMS replica, il gruppo di sottorete e l'attività, nonché un endpoint di AWS DMS origine e un endpoint di destinazione.

Per ulteriori informazioni su AMS AWS DMS, consulta le informazioni AWS Database Migration ServiceAWS Database Migration Service FAQsgenerali e le risposte alle domande più comuni.

AWS DMS, dati richiesti per la configurazione

Per ognuna delle seguenti AWS DMS procedure dettagliate, sono necessari alcuni dati in comune.

- `Description`: Informazioni significative sulla risorsa, separate dalle altre opzioni di parametro. `Description`
- `VpcId`: Il VPC da usare. Puoi scoprirlo eseguendo il ListVpcSummaries funzionamento dell'API SKMS (`list-vpc-summaries`nella CLI) o consultando VPCsla pagina nella console AMS. Per il riferimento all'API AMS SKMS, consulta la scheda Report nella console AWS Artifact.
- `Name`: Un nome per lo stack o il componente dello stack; questo diventa il nome dello stack.
- `TimeoutInMinutes`: Quanti minuti sono consentiti per la creazione dello stack prima che la RFC fallisca. Questa impostazione non ritarderà l'esecuzione RFC, ma è necessario concedere un periodo di tempo sufficiente (ad esempio, non specificare). `"5"`

- `ChangeTypeId`,`ChangeTypeVersion`, `eStackTemplateId`: Sono obbligatori ma variano in base al CT e i loro valori sono forniti in ciascuna sezione pertinente, di seguito.

AWS DMS attività di configurazione

Esegui la configurazione AWS DMS con le seguenti procedure dettagliate.

1: sottogruppo AWS DMS di replica: creazione

È possibile utilizzare la console AMS o API/CLI creare un sottogruppo di AWS DMS replica AMS.

Creare un sottogruppo di AWS DMS replica

Creazione di un gruppo di AWS DMS sottoreti di replica con la console

> **Note**
>
> Questo CT fallisce se il ruolo `dms-vpc-role` IAM non esiste nell'account.

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.

3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un Oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.
4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di un sottorete AWS DMS di replica con la CLI

> **Note**
>
> Questo CT fallisce se il ruolo `dms-vpc-role` IAM non esiste nell'account.

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-2q5azjd8p1ag5" --change-type-version "1.0" --title "TestDMSRepSG" --execution-
parameters "{\"Description\":\"DMSTestRepSG\",\"VpcId\":\"VPC-ID\",\"Name\":\"Test
 Stack\",\"Parameters\":{\"Description\":\"DESCRIPTION\",\"SubnetIds\":[\"SUBNET-ID\",
 \"SUBNET-ID\"]},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-j637f96ls1h4oy5fj
\"}"
```

CREAZIONE DEL MODELLO:

1. Esporta i parametri di esecuzione per questo tipo di modifica in un file JSON; questo esempio lo chiama CreateDmsRsgParams .json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2q5azjd8p1ag5" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsRsgParams.json
   ```

2. Modifica e salva il file .json dei parametri CreateDmsRsgParams di esecuzione. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

   ```
   {
   "Description":          "DMSTestRepSG",
   "VpcId":                "VPC_ID",
   "TimeoutInMinutes":     60,
   "StackTemplateId":      "stm-j637f96ls1h4oy5fj",
   "Name":                 "Test RSG",
   ```

```
"Parameters":   {
    "Description":            "DESCRIPTION",
    "SubnetIds":              ["SUBNET_ID", "SUBNET_ID"]
    }
}
```

3. Esporta il modello JSON in un file nella cartella corrente; questo esempio lo chiama CreateDmsRsgRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsRsgRfc.json
```

4. Modifica e salva il file.json. CreateDmsRsgRfc Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2q5azjd8p1ag5",
"Title":                "DMS-RSG-Create-RFC"
}
```

5. Create la RFC, specificando il file dei parametri di esecuzione e il CreateDmsRsgRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsRsgRfc.json --execution-
parameters file://CreateDmsRsgParams.json
```

   Nella risposta si riceve l'ID della nuova RFC e si può utilizzare per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

- Questo CT fallisce se il ruolo `dms-vpc-role` IAM non esiste nell'account.
- È possibile aggiungere fino a 50 tag, ma per farlo è necessario abilitare la visualizzazione di configurazione aggiuntiva.

  Per ulteriori informazioni sulle istanze di replica DMS e sui gruppi di sottoreti, vedere Configurazione di una rete per un'istanza di replica.

2: istanza AWS DMS di replica: Crea

È possibile utilizzare la console AMS o API/CLI creare un'istanza di AWS DMS replica AMS.

Crea un'istanza di AWS DMS replica

Creazione di un'istanza di AWS DMS replica con la console

Schermata di questo tipo di modifica nella console AMS:

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.
4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di un'istanza AWS DMS di replica con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-27apldkhqr0ol" --change-type-version "1.0" --title "TestDMSRepInstance" --
execution-parameters "{\"Description\":\"DMSTestRepInstance\",\"VpcId\":\"VPC-ID\",
\"Name\":\"REP-INSTANCE-NAME\",\"Parameters\":{\"InstanceClass\":\"dms.t2.micro\",
\"ReplicationSubnetGroupIdentifier\":\"TEST-REP-SG\",\"SecurityGroupIds\":\"SG-ID, SG-
ID\"},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-3n1j5hdrmiiiuqk6v\"}"
```

Mentre viene creata l'istanza di replica, è possibile specificare i datastore di origine e di destinazione. Gli archivi dati di origine e di destinazione possono trovarsi su un'istanza Amazon Elastic Compute Cloud (Amazon EC2), un bucket AWS S3, un'istanza DB Amazon Relational Database Service (Amazon RDS) o un database locale.

CREAZIONE DEL MODELLO:

1. Esporta i parametri di esecuzione per questo tipo di modifica in un file JSON; questo esempio lo chiama CreateDmsRiParams .json:

```
aws amscm get-change-type-version --change-type-id "ct-27apldkhqr0ol" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsRiParams.json
```

2. Modifica e salva il file .json dei parametri CreateDmsRiParams di esecuzione. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"Description":          "DMSTestRepInstance",
"VpcId":                "VPC_ID",
"Name":                 "Test RI",
"StackTemplateId":      "stm-3n1j5hdrmiiiuqk6v",
"TimeoutInMinutes":     60,
"Parameters":   {
    "Description":                      "DESCRIPTION",
    "InstanceClass":                    "dms.t2.micro",
    "ReplicationSubnetGroupIdentifier": "TEST-REP-SG",
    "SecurityGroupIds":                 ["SG-ID, SG-ID"}
    }
}
```

3. Esporta il modello JSON in un file nella cartella corrente; questo esempio lo chiama CreateDmsRiRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsRiRfc.json
```

4. Modifica e salva il file.json. CreateDmsRiRfc Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-27apldkhqr0ol",
"Title":                "DMS-RI-Create-RFC"
```

```
}
```

5. Create la RFC, specificando il file dei parametri di esecuzione e il CreateDmsRiRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsRiRfc.json --execution-
parameters file://CreateDmsRiParams.json
```

Nella risposta si riceve l'ID della nuova RFC e si può utilizzare per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

- È possibile aggiungere fino a 50 tag, ma per farlo è necessario abilitare la visualizzazione di configurazione aggiuntiva.
- È necessario creare un'istanza di replica su un' EC2 istanza nel VPC AMS che abbia una potenza di archiviazione e di elaborazione sufficiente per eseguire le attività assegnate e migrare i dati dal database di origine al database di destinazione. La dimensione richiesta di questa istanza varia a seconda della quantità di dati da migrare e delle attività che deve eseguire l'istanza. L'istanza di replica fornisce elevata disponibilità e supporto di failover utilizzando una distribuzione Multi-AZ quando si seleziona l'opzione. `MultiAZ` Per ulteriori informazioni sulle istanze di replica, consulta Working with an AWS DMS Replication Instance.

3: endpoint di AWS DMS origine: crea, crea per Mongo DB, crea per S3

Puoi utilizzare la console AMS o API/CLI creare un endpoint di origine AMS DMS per vari database, forniamo tre esempi.

Endpoint di origine DMS: creazione

Creazione di un endpoint di origine DMS con la console

Schermata di questo tipo di modifica nella console AMS:

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.

2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un Oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella visualizzazione Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.
4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di un endpoint di origine DMS con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

Monitora il comando RFC:. `aws amscm get-rfc --rfc-id ID`

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws --profile saml --region us-east-1 amscm create-rfc --title "MariaDB-DMS-
Source-Endpoint" --aws-account-id ACCOUNT-ID --change-type-id ct-0attesnjqy2cx --
change-type-version 1.0 --execution-parameters "{\"Description\":\"DESCRIPTION.\",
\"VpcId\":\"VPC-ID\",\"Name\":\"MariaDB-DMS-SE\",\"Parameters\":{\"EngineName\":
\"mariadb\",\"ServerName\":\"mariadb.db.example.com\",\"Port\":3306,\"Username\":
\"DB-USER\",\"Password\":\"DB-PW\"},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-
pud4ghhkp7395n9bc\"}"
```

CREAZIONE DEL MODELLO:

1. Esporta i parametri di esecuzione per questo tipo di modifica in un file JSON denominato CreateDmsSeParams .json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0attesnjqy2cx" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsSeParams.json
   ```

2. Modifica e salva il file JSON dei parametri di esecuzione. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"Description":          "MariaDB-DMS-SE",
"VpcId":                "VPC_ID",
"Name":                 "Test SE",
"StackTemplateId":      "stm-pud4ghhkp7395n9bc",
"TimeoutInMinutes":     60,
"Parameters":   {
    "Description":      "DESCRIPTION",
    "EngineName":       "mariadb",
    "ServerName":       "mariadb.db.example.com",
    "Port":             "3306",
    "Username":         "DB-USER",
    "Password":         "DB-PW",}
    }
}
```

3. Esporta il modello JSON in un file nella cartella corrente; questo esempio lo chiama CreateDmsSeRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsSeRfc.json
```

4. Modifica e salva il file.json. CreateDmsSeRfc Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0attesnjqy2cx",
"Title":                "MariaDB-DMS-Source-Endpoint"
}
```

5. Create la RFC, specificando il file dei parametri di esecuzione e il CreateDmsSeRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsSeRfc.json --execution-
parameters file://CreateDmsSeParams.json
```

Nella risposta si riceve l'ID della nuova RFC e si può utilizzare per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

Prima di creare l'endpoint DMS, assicurati che la password non contenga caratteri non supportati. Per ulteriori informazioni, consulta Creazione di endpoint di origine e destinazione nella Guida per l'utente.AWS Database Migration Service

Per ulteriori informazioni, consulta Sources for Data Migration.

Per un endpoint di origine S3, consulta. Endpoint di origine DMS per S3: creazione

Per un endpoint sorgente Mongo DB, vedi. Endpoint sorgente DMS per MongoDB: creazione

Endpoint sorgente DMS per MongoDB: creazione

Creazione di un endpoint sorgente DMS Mongo DB con la console

Schermata di questo tipo di modifica nella console AMS:

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un Oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella visualizzazione Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.

4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di un endpoint sorgente DMS Mongo DB con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws amscm  --profile saml --region us-east-1 create-rfc --change-type-id
 "ct-2hxcllf1b4ey0" --change-type-version "1.0" --title 'DMS_Source_MongoDB'
 --description "DESCRIPTION"  --execution-parameters "{\"Description\":
\"DMS_MongoDB_Source_Endpoint\",\"VpcId\":\"VPC_ID\",\"Name\":\"DMS-Mongo-SE\",
\"StackTemplateId\":\"stm-pud4ghhkp7395n9bc\",\"TimeoutInMinutes\":60,\"Parameters\":
{\"DatabaseName\":\"mytestdb\",\"EngineName\":\"mongodb\",\"Port\":27017,\"ServerName
\":\"test.example.com\"}}"
```

CREAZIONE DEL MODELLO:

1. Esporta i parametri di esecuzione per questo tipo di modifica in un file JSON denominato CreateDmsSeMongoParams .json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2hxcllf1b4ey0"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateDmsSeMongoParams.json
   ```

2. Modifica e salva il file JSON dei parametri di esecuzione. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

   ```
   {
   "Description":          "MongoDB-DMS-SE",
   "VpcId":                "VPC_ID",
   "StackTemplateId":      "stm-pud4ghhkp7395n9bc",
   "Name":                 "Test Mongo SE",
   "TimeoutInMinutes":     60,
   "Parameters":   {
       "Description":      "DESCRIPTION",
       "DatabaseName":       "mytestdb",
       "EngineName":       "mongodb",
       "ServerName":       "test.example.com",
       "Port":             "27017"
       }
   }
   ```

3. Esporta il modello JSON in un file nella cartella corrente; questo esempio lo chiama CreateDmsSeMongoRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsSeMongoRfc.json
```

4. Modifica e salva il file.json. CreateDmsSeMongoRfc Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2hxcllf1b4ey0",
"Title":                "DMS_Source_MongoDB"
}
```

5. Create la RFC, specificando il file dei parametri di esecuzione e il CreateDmsSeMongoRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsSeMongoRfc.json --execution-
parameters file://CreateDmsSeMongoParams.json
```

   Nella risposta si riceve l'ID della nuova RFC e si può utilizzare per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

> **Note**
>
> È possibile aggiungere fino a 50 tag, ma per farlo è necessario abilitare la visualizzazione di configurazione aggiuntiva.

AMS DMS può utilizzare Mongo o qualsiasi Relational Database Service (RDS) come endpoint di origine. Per un endpoint di origine S3, vedi. Endpoint di origine DMS per S3: creazione

Endpoint di origine DMS per S3: creazione

Creazione di un endpoint di origine DMS S3 con la console

Schermata di questo tipo di modifica nella console AMS:

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un Oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella visualizzazione Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.
4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di un endpoint di origine DMS S3 con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di

esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.

2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws --profile saml --region us-east-1 amscm create-rfc --title "S3DMSSourceEndpoint" --
aws-account-id ACCOUNT-ID --change-type-id ct-2oxl37nphsrjz --change-type-version 1.0
 --execution-parameters "{\"Description\":\"TestS3DMS-SE\",\"VpcId\":\"VPC-ID\",\"Name
\":\"S3-DMS-SE\",\"Parameters\":{\"EngineName\":\"s3\",\"S3BucketName\":\"amzn-s3-
demo-bucket\",\"S3ExternalTableDefinition\":\"{\\\"TableCount\\\":\\\"1\\\",\\\"Tables
\\\":[{\\\"TableName\\\":\\\"employee\\\",\\\"TablePath\\\":\\\"hr/employee/\\\",\\
\"TableOwner\\\":\\\"hr\\\",\\\"TableColumns\\\":[{\\\"ColumnName\\\":\\\"Id\\\",\\
\"ColumnType\\\":\\\"INT8\\\",\\\"ColumnNullable\\\":\\\"false\\\",\\\"ColumnIsPk\\\":
\\\"true\\\"},{\\\"ColumnName\\\":\\\"LastName\\\",\\\"ColumnType\\\":\\\"STRING\\\",
\\\"ColumnLength\\\":\\\"20\\\"},{\\\"ColumnName\\\":\\\"FirstName\\\",\\\"ColumnType
\\\":\\\"STRING\\\",\\\"ColumnLength\\\":\\\"30\\\"},{\\\"ColumnName\\\":\\\"HireDate\
\\",\\\"ColumnType\\\":\\\"DATETIME\\\"},{\\\"ColumnName\\\":\\\"OfficeLocation\\\",\\
\"ColumnType\\\":\\\"STRING\\\",\\\"ColumnLength\\\":\\\"20\\\"}],\\\"TableColumnsTotal
```

```
\\\":\\\"5\\\"}]}\",\"S3ServiceAccessRoleArn\":\"arn:aws:iam::123456789101:role/ams-
ops-ct-authors-dms-s3-test-role\"},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-
pud4ghhkp7395n9bc\"}"
```

CREAZIONE DEL MODELLO:

1. Esporta i parametri di esecuzione per questo tipo di modifica in un file JSON denominato CreateDmsSe S3Params.json.

```
aws amscm get-change-type-version --change-type-id "ct-2oxl37nphsrjz" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsSeS3Params.json
```

2. Modifica e salva il file JSON dei parametri di esecuzione. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"Description":          "TestS3DMS-SE",
"VpcId":                "VPC_ID",
"Name":                 "S3-DMS-SE",
"StackTemplateId":      "stm-pud4ghhkp7395n9bc",
"TimeoutInMinutes":     60,
"Parameters":   {
    "EngineName":               "s3",
    "S3BucketName":              "amzn-s3-demo-bucket",
    "S3ExternalTableDefinition":  "BUCKET-NAME",
    {"TableCount":                "1",
      "Tables":[{"TableName":"employee","TablePath":"hr/
employee/","TableOwner":"hr","TableColumns":
[{"ColumnName":"Id","ColumnType":"INT8","ColumnNullable":"false","ColumnIsPk":"true"},
{"ColumnName":"LastName","ColumnType":"STRING","ColumnLength":"20"},
{"ColumnName":"FirstName","ColumnType":"STRING","ColumnLength":"30"},
{"ColumnName":"HireDate","ColumnType":"DATETIME"},
{"ColumnName":"OfficeLocation","ColumnType":"STRING","ColumnLength":"20"}],"TableColumnsTot
    "S3ServiceAccessRoleArn":      "arn:aws:iam::123456789101:role/ams-ops-ct-
authors-dms-s3-test-role",
      }
}
```

3. Esporta il modello JSON in un file nella cartella corrente; questo esempio lo chiama CreateDmsSe s3rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsSeS3Rfc.json
```

4. Modificate e salvate il file S3RFC.json. CreateDmsSe Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2oxl37nphsrjz",
"Title":                "DMS_Source_S3"
}
```

5. Crea la RFC, specificando il file dei parametri di esecuzione e il file CreateDmsSe S3Rfc:

```
aws amscm create-rfc --cli-input-json file://CreateDmsSeS3Rfc.json --execution-
parameters file://CreateDmsSeS3Params.json
```

   Nella risposta si riceve l'ID della nuova RFC e si può utilizzare per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

> **Note**
>
> È possibile aggiungere fino a 50 tag, ma per farlo è necessario abilitare la visualizzazione di configurazione aggiuntiva.

AMS DMS può utilizzare S3 o qualsiasi endpoint di origine Relational Database Service (RDS). Per un endpoint di origine Mongo DB, vedi. Endpoint sorgente DMS per MongoDB: creazione

4: endpoint di AWS DMS destinazione: crea, crea per S3

Puoi utilizzare la console AMS o API/CLI creare un endpoint di destinazione AMS DMS per vari database, forniamo due esempi.

Endpoint di destinazione DMS: creazione

AMS DMS può utilizzare S3 o qualsiasi Relational Database Service (RDS) con MySQL, MariaDB, Oracle, Postgresql o Microsoft SQL come endpoint di destinazione.

Creazione di un endpoint DMS Target con la console

Schermata di questo tipo di modifica nella console AMS:

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.
4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di un endpoint di destinazione DMS con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-3gf8dolbo8x9p" --change-type-version "1.0" --title "TestDMSTargetEndpoint" --
execution-parameters "{\"Description\":\"TestTE\",\"VpcId\":\"VPC-ID\",\"Name\":
\"TE-NAME\",\"StackTemplateId\":\"stm-knghtmmgefafdq89u\",\"TimeoutInMinutes\":60,
\"Parameters\":{\"EngineName\":\"mysql\",\"Password\":\"testpw123\",\"Port\":\"3306\",
\"ServerName\":\"mytestdb.d5fga0rf2wpi.ap-southeast-2.rds.amazonaws.com\",\"Username\":
\"USERNAME\"}}"
```

CREAZIONE DEL MODELLO:

1. Esporta i parametri di esecuzione per questo tipo di modifica in un file JSON denominato CreateDmsTeParams .json.

```
aws amscm get-change-type-version --change-type-id "ct-3gf8dolbo8x9p" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsTeParams.json
```

2. Modifica e salva il file JSON dei parametri di esecuzione. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"Description":          "TestTE",
"VpcId":                "VPC_ID",
"StackTemplateId":      "stm-knghtmmgefafdq89u",
"Name":                 "TE-NAME",
"TimeoutInMinutes":     60,
"Parameters":   {
    "EngineName":       "mysql",
    "ServerName":       "sql.db.example.com",
    "Port":             "3306",
    "Username":         "DB-USER",
    "Password":         "DB-PW",}
    }
}
```

3. Esporta il modello JSON in un file nella cartella corrente; questo esempio lo chiama CreateDmsTeRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsTeRfc.json
```

4. Modifica e salva il file.json. CreateDmsTeRfc Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3gf8dolbo8x9p",
"Title":                "DB-DMS-Target-Endpoint"
}
```

5. Create la RFC, specificando il file dei parametri di esecuzione e il CreateDmsTeRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsTeRfc.json --execution-
parameters file://CreateDmsTeParams.json
```

Nella risposta si riceve l'ID della nuova RFC e si può utilizzare per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

- Questo tipo di modifica è ora alla versione 2.0.
- AMS DMS può utilizzare S3 o qualsiasi Relational Database Service (RDS) con MySQL, MariaDB, Oracle, Postgresql o Microsoft SQL come endpoint di destinazione. Per un Endpoint di destinazione DMS per S3: creazione endpoint di destinazione S3, vedi.
- Per ulteriori informazioni, consulta Target for Data Migration.
- Puoi aggiungere fino a 50 tag, ma per farlo devi abilitare la visualizzazione di configurazione aggiuntiva.

Endpoint di destinazione DMS per S3: creazione

Creazione di un endpoint DMS S3 Target con la console

Schermata di questo tipo di modifica nella console AMS:

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina

Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.

- Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.

3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.

4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.

5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di un endpoint di destinazione DMS S3 con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-05muqzievnxk5" --change-type-version "1.0" --title "TestDMSTargetEndpointS3"
 --execution-parameters "{\"Description\":\"TestS3TE\",\"VpcId\":\"VPC-ID\",\"Name
\":\"S3TE-NAME\",\"StackTemplateId\":\"stm-knghtmmgefafdq89u\",\"TimeoutInMinutes
\":60,\"Parameters\":{\"EngineName\":\"s3\",\"S3BucketName\":\"amzn-s3-demo-bucket\",
\"S3ServiceAccessRoleArn\":\"arn:aws:iam::123456789123:role/my-s3-role\"}}"
```

CREAZIONE DEL MODELLO:

1. Esporta i parametri di esecuzione per questo tipo di modifica in un file JSON; questo esempio lo chiama CreateDmsTe s3Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-05muqzievnxk5" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsTeS3Params.json
   ```

2. Modifica e salva il file S3Params.json dei parametri di esecuzione. CreateDmsTe Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

   ```
   {
   "Description":          "TestS3DMS-TE",
   "VpcId":                "VPC_ID",
   "StackTemplateId":      "stm-knghtmmgefafdq89u",
   "Name":                 "DMS-S3-TE",
   "TimeoutInMinutes":     60,
   ```

```
"Parameters":   {
    "EngineName":       "s3",
    "S3BucketName":      "amzn-s3-demo-bucket",
    "S3ServiceAccessRoleArn":       "arn:aws:iam::123456789101:role/ams-ops-ct-
authors-dms-s3-test-role"
      }
}
```

3. Esporta il modello JSON in un file nella cartella corrente; questo esempio lo chiama CreateDmsTe s3rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsTeS3Rfc.json
```

4. Modificate e salvate il file S3RFC.json. CreateDmsTe Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-05muqzievnxk5",
"Title":                "DMS_Target_S3"
}
```

5. Crea la RFC, specificando il file dei parametri di esecuzione e il file CreateDmsTe S3Rfc:

```
aws amscm create-rfc --cli-input-json file://CreateDmsTeS3Rfc.json --execution-
parameters file://CreateDmsTeS3Params.json
```

   Nella risposta si riceve l'ID della nuova RFC e si può utilizzare per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

> **Note**
>
> È possibile aggiungere fino a 50 tag, ma per farlo è necessario abilitare la visualizzazione di configurazione aggiuntiva.

AMS fornisce un tipo di modifica separato per la creazione di un endpoint di destinazione per S3. Per ulteriori informazioni, consulta Utilizzo di Amazon S3 come destinazione per AWS Database

Migration Service e attributi di connessione aggiuntivi quando si utilizza Amazon S3 come destinazione per AWS DMS.

5: attività di AWS DMS replica: creazione

È possibile utilizzare la console AMS o API/CLI creare un'attività di AWS DMS replica AMS.

Creare un'attività di AWS DMS replica

Creazione di un'attività AWS DMS di replica con la console

Schermata di questo tipo di modifica nella console AMS:

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un Oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.

4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Creazione di un'attività AWS DMS di replica con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-1d2fml15b9eth" --change-type-version "1.0" --title "TestDMSRepTask" --
execution-parameters "{\"Description\":\"TestRepTask\",\"VpcId\":\"VPC-ID\",\"Name
\":\"DMSRepTask\",\"Parameters\":{\"CdcStartTime\":\1533776569\"MigrationType\":
\"full-load\",\"ReplicationInstanceArn\":\"REP_INSTANCE_ARN\",\"SourceEndpointArn
\":\"SOURCE_ENDPOINT_ARN\",\"TableMappings\":\"{\\\"rules\\\": [{\\\"rule-type
\\\": \\\"selection\\\",\\\"rule-id\\\": \\\"1\\\",\\\"rule-name\\\": \\\"1\\
\",\\\"object-locator\\\": {\\\"schema-name\\\": \\\"Test\\\",\\\"table-name\\
\": \\\"%\\\"}, \\\"rule-action\\\": \\\"include\\\"}] }\",\"TargetEndpointArn
\":\"TARGET_ENDPOINT_ARN\"},\"StackTemplateId\":\"stm-eos7uq0usnmeggdet\",
\"TimeoutInMinutes\":60}"
```

CREAZIONE DEL MODELLO:

1. Esporta i parametri di esecuzione per questo tipo di modifica in un file JSON; questo esempio lo chiama CreateDmsRtParams .json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1d2fml15b9eth" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsRtParams.json
   ```

2. Modifica e salva il file JSON dei parametri di esecuzione. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

   ```
   {
   "Description":          "DMSTestRepTask",
   "VpcId":                "VPC_ID",
   "StackTemplateId":      "stm-eos7uq0usnmeggdet",
   "Name":                 "Test DMS RT",
   "TimeoutInMinutes":     60,
   "Parameters":   {
       "CdcStartTime":           "1533776569",
       "MigrationType":          "full-load",
       "ReplicationInstanceArn": "REP_INSTANCE_ARN",
       "SourceEndpointArn":      "SOURCE_ENDPOINT_ARN",
       "TargetEndpointArn":      "TARGET_ENDPOINT_ARN"
       "TableMappings":          {"rules": [{"rule-type": "selection","rule-id":
    "1","rule-name": "1","object-locator": {"schema-name": "Test","table-name": "%"},
    "rule-action": "include"}] }",
       }
   }
   ```

3. Esporta il modello JSON in un file nella cartella corrente; questo esempio lo chiama CreateDmsRtRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsRtRfc.json
```

4. Modifica e salva il file.json. CreateDmsRtRfc Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1d2fml15b9eth",
"Title":                "DMS-RI-Create-RFC"
}
```

5. Create la RFC, specificando il file dei parametri di esecuzione e il CreateDmsRtRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsRtRfc.json --execution-
parameters file://CreateDmsRtParams.json
```

   Nella risposta si riceve l'ID della nuova RFC e si può utilizzare per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

È possibile creare un' AWS DMS attività che acquisisca tre diversi tipi di modifiche o dati. Per ulteriori informazioni, consulta Working with AWS DMS Tasks, Creazione di un'attività e Creazione di attività per la replica continua con AWS DMS.

AWS DMS gestione

AWS DMS esempi di gestione.

Avviare l' AWS DMS attività di replica

Avvio di un'attività di AWS DMS replica con la console

Schermata di questo tipo di modifica nella console AMS:

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

   Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.
4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Avvio di un'attività AWS DMS di replica con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di

esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.

2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws amscm create-rfc --change-type-id "ct-1yq7hhqse71yg" --change-type-version
 "1.0" --title "Start DMS Replication Task" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-StartDmsTask\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ReplicationTaskArn\":[\"TASK_ARN\"],\"StartReplicationTaskType\":[\"start-
replication\"],\"CdcStartPosition\":[\"\"],\"CdcStopPosition\":[\"\"]}}"
```

CREAZIONE DEL MODELLO:

1. Esporta i parametri di esecuzione per questo tipo di modifica in un file JSON; questo esempio lo chiama StartDmsRtParams .json:

```
aws amscm get-change-type-version --change-type-id "ct-1yq7hhqse71yg" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > StartDmsRtParams.json
```

2. Modifica e salva il file JSON dei parametri di esecuzione. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
  "DocumentName": "AWSManagedServices-StartDmsTask",
  "Region": "us-east-1",
  "Parameters": {
      "ReplicationTaskArn": [
        "TASK_ARN"
      ],
      "StartReplicationTaskType": [
        "start-replication"
      ],
      "CdcStartPosition": [
        ""
      ],
      "CdcStopPosition": [
        ""
      ]
  }
}
```

3. Esporta il modello JSON in un file nella cartella corrente; questo esempio lo chiama StartDmsRtRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > StartDmsRtRfc.json
```

4. Modifica e salva il file.json. StartDmsRtRfc Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
  "ChangeTypeId": "ct-1yq7hhqse71yg",
  "ChangeTypeVersion": "1.0",
  "Title": "Start DMS Replication Task"
}
```

5. Create la RFC, specificando il file dei parametri di esecuzione e il StartDmsRtRfc file:

```
aws amscm create-rfc --cli-input-json file://StartDmsRtRfc.json --execution-
parameters file://StartDmsRtParams.json
```

Nella risposta si riceve l'ID della nuova RFC e si può utilizzare per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

È possibile avviare un'attività di AWS DMS replica utilizzando la console AMS o l'API/CLI AMS. Per ulteriori informazioni, consulta Working with AWS DMS Tasks.

Interrompi l'attività di AWS DMS replica

Interruzione di un' AWS DMS attività di replica con la console

Schermata di questo tipo di modifica nella console AMS:

Come funziona:

1. Vai alla pagina Crea RFC: nel riquadro di navigazione a sinistra della console AMS, fai clic RFCsper aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.
2. Scegli un tipo di modifica (CT) popolare nella visualizzazione predefinita Sfoglia i tipi di modifica o seleziona un CT nella visualizzazione Scegli per categoria.
   - Naviga per tipo di modifica: puoi fare clic su un CT popolare nell'area di creazione rapida per aprire immediatamente la pagina Run RFC. Nota che non puoi scegliere una versione CT precedente con creazione rapida.

     Per ordinare CTs, utilizzate l'area Tutti i tipi di modifica nella vista a scheda o tabella. In entrambe le visualizzazioni, selezionate un CT, quindi fate clic su Crea RFC per aprire la pagina Esegui RFC. Se applicabile, accanto al pulsante Crea RFC viene visualizzata l'opzione Crea con una versione precedente.
   - Scegli per categoria: seleziona una categoria, sottocategoria, articolo e operazione e la casella dei dettagli CT si apre con l'opzione Crea con una versione precedente, se applicabile. Fai clic su Crea RFC per aprire la pagina Esegui RFC.
3. Nella pagina Run RFC, apri l'area del nome CT per visualizzare la casella dei dettagli CT. È richiesto un oggetto (questo campo viene compilato automaticamente se si sceglie il CT nella

vista Sfoglia i tipi di modifica). Apri l'area di configurazione aggiuntiva per aggiungere informazioni sull'RFC.

Nell'area di configurazione dell'esecuzione, utilizza gli elenchi a discesa disponibili o inserisci i valori per i parametri richiesti. Per configurare i parametri di esecuzione opzionali, aprite l'area di configurazione aggiuntiva.

4. Al termine, fate clic su Esegui. Se non sono presenti errori, viene visualizzata la pagina RFC creata correttamente con i dettagli RFC inviati e l'output iniziale di Run.
5. Apri l'area dei parametri di esecuzione per visualizzare le configurazioni inviate. Aggiorna la pagina per aggiornare lo stato di esecuzione RFC. Facoltativamente, annulla la RFC o creane una copia con le opzioni nella parte superiore della pagina.

Interruzione di un'attività di AWS DMS replica con la CLI

Come funziona:

1. Usa Inline Create (esegui un `create-rfc` comando con tutti i parametri RFC e di esecuzione inclusi) o Template Create (crei due file JSON, uno per i parametri RFC e uno per i parametri di esecuzione) ed esegui il `create-rfc` comando con i due file come input. Entrambi i metodi sono descritti qui.
2. Invia il `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con l'ID RFC restituito.

   Monitora il comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Per verificare la versione del tipo di modifica, usa questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> È possibile utilizzare qualsiasi `CreateRfc` parametro con qualsiasi RFC, indipendentemente dal fatto che faccia parte o meno dello schema per il tipo di modifica. Ad esempio, per ricevere notifiche quando lo stato RFC cambia, aggiungi questa riga `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` alla

parte dei parametri RFC della richiesta (non ai parametri di esecuzione). Per un elenco di tutti i CreateRfc parametri, consulta l'AMS Change Management API Reference.

CREAZIONE IN LINEA:

Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea), quindi invia l'ID RFC restituito. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
aws amscm create-rfc --change-type-id "ct-1vd3y4ygbqmfk" --change-type-version
 "1.0" --title "Stop DMS Replication Task" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-StopDmsTask\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ReplicationTaskArn\":[\"TASK_ARN\"]}}"
```

CREAZIONE DEL MODELLO:

1. Esporta i parametri di esecuzione per questo tipo di modifica in un file JSON; questo esempio lo chiama StopDmsRtParams .json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1vd3y4ygbqmfk" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StopDmsRtParams.json
   ```

2. Modifica e salva il file JSON dei parametri di esecuzione. Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

   ```
   {
     "DocumentName": "AWSManagedServices-StopDmsTask",
     "Region": "us-east-1",
     "Parameters": {
         "ReplicationTaskArn": [
           "TASK_ARN"
         ]
     }
   }
   ```

3. Esporta il modello JSON in un file nella cartella corrente; questo esempio lo chiama StopDmsRtRfc .json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > StopDmsRtRfc.json
   ```

4. Modifica e salva il file.json. StopDmsRtRfc Ad esempio, puoi sostituire il contenuto con qualcosa del genere:

```
{
  "ChangeTypeId": "ct-1vd3y4ygbqmfk",
  "ChangeTypeVersion": "1.0",
  "Title": "Stop DMS Replication Task"
}
```

5. Create la RFC, specificando il file dei parametri di esecuzione e il StopDmsRtRfc file:

```
aws amscm create-rfc --cli-input-json file://StopDmsRtRfc.json --execution-
parameters file://StopDmsRtParams.json
```

   Nella risposta si riceve l'ID della nuova RFC e si può utilizzare per inviare e monitorare la RFC. Finché non la invii, la RFC rimane nello stato di modifica e non si avvia.

Suggerimenti

È possibile interrompere un'attività di replica DMS utilizzando la console AMS o l'API/CLI AMS. Per ulteriori informazioni, consulta Working with AWS DMS Tasks.

Importazione di database (DB) su AMS RDS per Microsoft SQL Server

> **Note**
>
> Gli endpoint AMS API/CLI (amscm e amsskms) si trovano nella regione AWS della Virginia settentrionale,. `us-east-1` A seconda di come è impostata l'autenticazione e della regione AWS in cui si trovano l'account e le risorse, potrebbe essere necessario aggiungerli `--region us-east-1` quando si emettono i comandi. Potrebbe anche essere necessario aggiungere`--profile saml`, se questo è il metodo di autenticazione utilizzato.

Il processo di importazione DB in AMS RDS per SQL Server si basa sui tipi di modifica AMS (CTs) inviati come richieste di modifica (RFCs) e utilizza i parametri dell'API Amazon RDS come input. MicroSoft SQL Server è un sistema di gestione di database relazionali (RDBMS). Per ulteriori

informazioni, consulta anche: Amazon Relational Database Service (Amazon RDS) e riferimento alle API rds o Amazon RDS.

> **Note**
>
> Assicurati che ogni RFC sia stata completata correttamente prima di passare alla fase successiva.

Fasi di importazione di alto livello:

1. Esegui il backup del database MS SQL di origine locale in un file.bak (backup)
2. Copia il file.bak nel bucket di transito (crittografato) Amazon Simple Storage Service (S3)
3. Importa il file.bak in un nuovo DB sull'istanza Amazon RDS MS SQL di destinazione

Requisiti:

- Stack MS SQL RDS in AMS
- Stack RDS con opzione di ripristino () `SQLSERVER_BACKUP_RESTORE`
- Secchio Transit S3
- Ruolo IAM con accesso al bucket che consente ad Amazon RDS di assumere il ruolo
- Un' EC2 istanza con MS SQL Management Studio installato per gestire RDS (può essere una workstation locale)

Configurazione

Completa queste attività per iniziare il processo di importazione.

1. Invia una RFC per creare uno stack RDS utilizzando Deployment | Advanced stack components | RDS database stack | Create (ct-2z60dyvto9g6c). Non utilizzare il nome (*`RDSDBName`*parametro) del DB di destinazione nella richiesta di creazione, il DB di destinazione verrà creato durante l'importazione. Assicurati di lasciare spazio sufficiente (`RDSAllocatedStorage`parametro). Per maggiori dettagli su questa operazione, consulta la AMS Change Management Guide RDS DB Stack | Create.

2. Invia una RFC per creare il bucket S3 di transito (se non esiste già) utilizzando Deployment | Advanced stack components | S3 storage | Create (ct-1a68ck03fn98r). Per maggiori dettagli su questa operazione, consulta l'AMS Change Management Guide S3 Storage | Create.
3. Invia una RFC Management | Other | Other | Update (ct-1e1xtak34nx76) per implementarla con questi dettagli: `customer_rds_s3_role`

   Nella console:

   - Oggetto: «Per supportare l'importazione di database da MS SQL Server, esegui l'implementazione su questo account. `customer_rds_s3_role`
   - Nome del bucket Transit S3:. *`BUCKET_NAME`*
   - Informazioni di contatto:. *`EMAIL`*

   Con un ImportDbParams file.json per la CLI:

   ```
   {
          "Comment": "{"Transit S3 bucket name":"BUCKET_NAME"}",
          "Priority": "High"
        }
   }
   ```

4. Invia un Management | Other | Other | Aggiorna RFC richiedendo ad AMS di impostare l'`SQLSERVER_BACKUP_RESTORE`opzione sull'RDS creato nel passaggio 1 (utilizza l'ID dello stack dall'output della fase 1 e il ruolo `customer_rds_s3_role` IAM in questa richiesta, in questa richiesta).
5. Invia una RFC per creare un' EC2 istanza (puoi usare qualsiasi postazione/istanza esistente EC2 o locale) e installa Microsoft SQL Management Studio sull'istanza.

Importazione del database

Per importare il database (DB), segui questi passaggi.

1. Esegui il backup del database locale di origine utilizzando il backup e il ripristino nativi di MS SQL (vedi Supporto per il backup e il ripristino nativi in SQL Server). Come risultato dell'esecuzione di tale operazione, dovresti avere un file.bak (backup).
2. Carica il file.bak in un bucket S3 di transito esistente utilizzando la CLI AWS S3 o la console AWS S3. Per informazioni sui bucket S3 in transito, consulta Protezione dei dati tramite crittografia.

3. Importa il file.bak in un nuovo DB sull'istanza di destinazione RDS for SQL Server MS SQL (per dettagli sui tipi, consulta Amazon RDS for MySQL Instance types):

   a. Accedi all' EC2 istanza (workstation locale) e apri MS SQL Management Studio

   b. Connect all'istanza RDS di destinazione creata come prerequisito nel passaggio #1. Segui questa procedura per connetterti: Connessione a un'istanza DB che esegue il motore di database Microsoft SQL Server

   c. Avvia il processo di importazione (ripristino) con una nuova query SQL (Structured Query Language) (per i dettagli sulle query SQL, vedere Introduzione a SQL). Il nome del database di destinazione deve essere nuovo (non utilizzare lo stesso nome del database creato in precedenza). Esempio senza crittografia:

   ```
   exec msdb.dbo.rds_restore_database
           @restore_db_name=TARGET_DB_NAME,

           @s3_arn_to_restore_from='arn:aws:s3:::BUCKET_NAME/FILENAME.bak';
   ```

   d. Controlla periodicamente lo stato del processo di importazione eseguendo questa query in una finestra separata:

   ```
   exec msdb.dbo.rds_task_status;
   ```

   Se lo stato cambia in Non riuscito, cerca i dettagli dell'errore nel messaggio.

Rimozione

Dopo aver importato il database, potresti voler rimuovere le risorse non necessarie, segui questi passaggi.

1. Elimina il file di backup (.bak) dal bucket S3. A tale scopo, puoi utilizzare la console S3. Per il comando CLI per eliminare un oggetto da un bucket S3, consulta rm nell'AWS CLI Command Reference.
2. Elimina il bucket S3 se non hai intenzione di utilizzarlo. Per i passaggi da seguire, consulta Delete Stack.
3. Se non hai intenzione di effettuare importazioni da MS SQL, invia un RFC Management | Other | Other | Update (ct-0xdawir96cy7k) e richiedi che AMS elimini il ruolo IAM. `customer_rds_s3_role`

Implementazioni di app Tier e Tie in AMS

Una distribuzione Tier and Tie consente di creare, configurare e distribuire le risorse di uno stack in modo indipendente utilizzando componenti separati RFCs e di utilizzare i componenti IDs dello stack man mano che si procede per associarli tra loro.

Ad esempio, per implementare un sito Web ad alta disponibilità (ridondante) con un sistema di bilanciamento del carico e un database, utilizzando un approccio Tier and Tie, invia RFCs un database e un sistema di bilanciamento del carico e due istanze EC2 o un gruppo Auto Scaling e configura le istanze EC2 o il gruppo Auto Scaling con l'ID dell'ELB che hai creato.

Dopo la distribuzione delle risorse, puoi inviare un gruppo di sicurezza create change per consentire alle risorse di comunicare con il database. Per i dettagli sulla creazione di gruppi di sicurezza, consulta Create Security Group.

Implementazioni complete di app in AMS

Una distribuzione Full Stack consiste nell'inviare una RFC con un CT che crea e configura tutto ciò di cui hai bisogno contemporaneamente. Ad esempio, per implementare il sito Web ad alta disponibilità appena descritto (EC2 istanze, sistema di bilanciamento del carico e database) dovreste utilizzare un CT che, insieme, creasse e configurasse un gruppo di Auto Scaling, un sistema di bilanciamento del carico, un database e le impostazioni del gruppo di sicurezza necessarie affinché tutte le istanze funzionino come uno stack. Di seguito vengono descritti alcuni esempi di due AMS CTs che eseguono questa operazione.

- Stack a due livelli ad alta disponibilità (ct-06mjngx5flwto): questo tipo di modifica consente di creare uno stack e configurare un gruppo di Auto Scaling, un database supportato da RDS, un Load Balancer, un'applicazione e una configurazione. CodeDeploy Tieni presente che il load balancer non è considerato un tier in quanto è condiviso tra più applicazioni come appliance di rete e anche le funzioni sono considerate un'appliance. CodeDeploy Inoltre, crea un gruppo di CodeDeploy distribuzione (con il nome assegnato all' CodeDeploy applicazione) che può essere utilizzato per distribuire le applicazioni. Le impostazioni dei gruppi di sicurezza per consentire alle risorse di funzionare insieme vengono create automaticamente.
- High Availability One-Tier Stack (ct-09t6q7j9v5hrn): questo tipo di modifica consente di creare uno stack e configurare un Auto Scaling Group e un Application Load Balancer. Le impostazioni dei gruppi di sicurezza che consentono alle risorse di funzionare insieme vengono create automaticamente.

Utilizzo dei tipi di modifica del provisioning () CTs

AMS è responsabile dell'infrastruttura gestita, per apportare modifiche è necessario inviare una RFC con la classificazione CT corretta (categoria, sottocategoria, articolo e operazione). Questa sezione descrive come trovarne CTs, determinare se una TAC è adatta alle proprie esigenze e richiedere una nuova TC se nessuna lo è.

Verifica se una TAC esistente soddisfa i tuoi requisiti

Dopo aver determinato cosa vuoi implementare con AMS, il passo successivo consiste nello studiare i CloudFormation modelli esistenti CTs e quelli esistenti per vedere se esiste già una soluzione.

Quando si crea un RFC, è necessario specificare il CT. È possibile utilizzare AWS Management Console o l'API/CLI AMS. Di seguito vengono descritti alcuni esempi di utilizzo di entrambi.

È possibile utilizzare la console o il API/CLI per trovare una modifica del tipo di ID (CT) o della versione. Esistono due metodi: la ricerca o la scelta della classificazione. Per entrambi i tipi di selezione, è possibile ordinare la ricerca scegliendo Usato più di frequente, Usato più di recente o Alfabetico.

YouTube Video: Come posso creare una RFC utilizzando la CLI di AWS Managed Services e dove posso trovare lo schema CT?

Nella console AMS, nella RFCspagina -> Crea RFC:

- Con l'opzione Sfoglia per tipo di modifica selezionata (impostazione predefinita), puoi:
  - Utilizza l'area di creazione rapida per selezionare tra i più diffusi di AMS CTs. Fai clic su un'etichetta e si apre la pagina Esegui RFC con l'opzione Oggetto compilata automaticamente per te. Completa le opzioni rimanenti secondo necessità e fai clic su Esegui per inviare la RFC.
  - In alternativa, scorri verso il basso fino all'area Tutti i tipi di modifica e inizia a digitare un nome CT nella casella delle opzioni, non è necessario avere il nome esatto o completo del tipo di modifica. Puoi anche cercare un CT in base alla modifica dell'ID del tipo, alla classificazione o alla modalità di esecuzione (automatica o manuale) inserendo le parole pertinenti.

    Con la visualizzazione Carte predefinita selezionata, le schede CT corrispondenti vengono visualizzate durante la digitazione, seleziona una scheda e fai clic su Crea RFC. Con la vista Tabella selezionata, scegli il CT pertinente e fai clic su Crea RFC. Entrambi i metodi aprono la pagina Run RFC.

- In alternativa, e per esplorare le opzioni di modifica, fai clic su Scegli per categoria nella parte superiore della pagina per aprire una serie di caselle di opzioni a discesa.
- Scegli una categoria, una sottocategoria, un articolo e un'operazione. Viene visualizzata la casella delle informazioni per quel tipo di modifica e nella parte inferiore della pagina viene visualizzato un pannello.
- Quando sei pronto, premi Invio per visualizzare un elenco dei tipi di modifica corrispondenti.
- Scegli un tipo di modifica dall'elenco. La casella delle informazioni per quel tipo di modifica viene visualizzata nella parte inferiore della pagina.
- Dopo aver impostato il tipo di modifica corretto, scegli Crea RFC.

> **Note**
>
> Affinché questi comandi funzionino, è necessario installare l'AMS CLI. Per installare l'API o la CLI AMS, vai alla pagina Risorse per gli sviluppatori della console AMS. Per materiale di riferimento sull'API AMS CM o sull'API AMS SKMS, consulta la sezione AMS Information Resources nella Guida per l'utente. Potrebbe essere necessario aggiungere un'`--profile`opzione per l'autenticazione, `aws amsskms` *`ams-cli-command`* `--profile SAML` ad esempio. Potrebbe essere necessario aggiungere l'`--region`opzione anche perché tutti i comandi AMS non utilizzano us-east-1, ad esempio. `aws amscm` *`ams-cli-command`* `--region=us-east-1`

> **Note**
>
> Gli endpoint AMS API/CLI (amscm e amsskms) si trovano nella regione AWS della Virginia settentrionale,. `us-east-1` A seconda di come è impostata l'autenticazione e della regione AWS in cui si trovano l'account e le risorse, potrebbe essere necessario aggiungerli `--region us-east-1` quando si emettono i comandi. Potrebbe anche essere necessario aggiungere`--profile saml`, se questo è il metodo di autenticazione utilizzato.

Per cercare un tipo di modifica utilizzando l'API AMS CM (vedi ListChangeTypeClassificationSummaries) o la CLI:

Puoi utilizzare un filtro o una query per effettuare la ricerca. L' ListChangeTypeClassificationSummaries operazione dispone delle opzioni di filtro per

`CategorySubcategory`,`Item`, e`Operation`, ma i valori devono corrispondere esattamente ai valori esistenti. Per risultati più flessibili quando si utilizza la CLI, è possibile utilizzare l'`--query`opzione.

Filtraggio dei cambi di tipo con l'API/CLI AMS CM

Attributo	Valori validi	Condizione valida/predefinita	Note
ChangeTypeId	Qualsiasi stringa che rappresenta un ChangeTypeId (ad esempio: ct-abc123xyz7890)	Equals	Per il tipo di modifica, vedere il riferimento al tipo di modifica. IDs Per il tipo di modifica IDs, vedere Finding a Change Type o CSIO.
Categoria	Qualsiasi testo in formato libero	Contiene	Le espressioni regolari in ogni singolo campo non sono supportate. Ricerca senza distinzione tra lettere maiuscole
Sottocategoria			
Elemento			
Operazione			

1. Ecco alcuni esempi di classificazione dei tipi di modifica delle inserzioni:

   Il comando seguente elenca tutte le categorie dei tipi di modifica.

   ```
   aws amscm list-change-type-categories
   ```

   Il comando seguente elenca le sottocategorie appartenenti a una categoria specificata.

   ```
   aws amscm list-change-type-subcategories --category CATEGORY
   ```

   Il comando seguente elenca gli elementi appartenenti a una categoria e sottocategoria specificate.

```
aws amscm list-change-type-items --category CATEGORY --subcategory SUBCATEGORY
```

2. Ecco alcuni esempi di ricerca dei tipi di modifica con le query CLI:

   Il comando seguente cerca nei riepiloghi delle classificazioni CT quelli che contengono «S3" nel nome dell'elemento e crea l'output della categoria, della sottocategoria, dell'elemento, dell'operazione e dell'ID del tipo di modifica sotto forma di tabella.

```
aws amscm list-change-type-classification-summaries --query
 "ChangeTypeClassificationSummaries [?contains(Item, 'S3')].
[Category,Subcategory,Item,Operation,ChangeTypeId]" --output table
```

```
+-------------------------------------------------------------+
|               ListChangeTypeClassificationSummaries         |
+----------+-------------------------+--+------+--------------+
|Deployment|Advanced Stack Components|S3|Create|ct-1a68ck03fn98r|
+----------+-------------------------+--+------+--------------+
```

3. È quindi possibile utilizzare l'ID del tipo di modifica per ottenere lo schema CT ed esaminare i parametri. Il comando seguente genera lo schema in un file JSON denominato Creates3Params.schema.json.

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateS3Params.schema.json
```

   Per informazioni sull'utilizzo delle query CLI, vedere Come filtrare l'output con l'opzione --query e il riferimento al linguaggio di interrogazione, Specificazione. JMESPath

4. Dopo aver ottenuto l'ID del tipo di modifica, ti consigliamo di verificare la versione del tipo di modifica per assicurarti che sia la versione più recente. Usa questo comando per trovare la versione per un tipo di modifica specificato:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CHANGE_TYPE_ID
```

   Per trovare la versione `AutomationStatus` per un tipo di modifica specifico, esegui questo comando:

```
aws amscm --profile saml get-change-type-version --change-type-id CHANGE_TYPE_ID --
query "ChangeTypeVersion.{AutomationStatus:AutomationStatus.Name}"
```

Per trovare il tipo di modifica `ExpectedExecutionDurationInMinutes` per un tipo di modifica specifico, esegui questo comando:

```
aws amscm --profile saml get-change-type-version --change-type-id ct-14027q0sjyt1h
 --query "ChangeTypeVersion.{ExpectedDuration:ExpectedExecutionDurationInMinutes}"
```

Una volta trovato un CT che ritieni appropriato, esamina lo schema JSON dei parametri di esecuzione ad esso associato per scoprire se è adatto al tuo caso d'uso.

Utilizzate questo comando per generare uno schema CT in un file JSON che prende il nome dal CT; questo esempio restituisce lo schema di archiviazione Create S3:

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
--query "ChangeTypeVersion.ExecutionInputSchema" --output text >
CreateBucketParams.json
```

Diamo un'occhiata più da vicino a ciò che offre questo schema.

S3 Bucket Crea schema

```
{
  "$schema": "http://json-schema.org/draft-04/sch
ema#",
"name": "Create S3 Storage
"description": "Use to create an Amazon Simple
 Storage Service stack.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "The description of the
 stack.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
```

Lo schema inizia con CT («descrizione»), che indica a cosa serve lo schema. In questo caso, per creare uno stack di archiviazione S3.

Successivamente, sono disponibili proprietà obbligatorie e facoltative che è possibile specificare. Vengono forniti i valori predefiniti delle proprietà . Le proprietà richieste sono elencate alla fine dello schema.

```
      "description": "ID of the VPC to create the S3
 Bucket in, in the form vpc-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{17}$"
    },
    "StackTemplateId": {
      "description": "Required value: stm-s2b72
beb000000000.",
      "type": "string",
      "enum": ["stm-s2b72beb000000000"]
    },
    "Name":{
      "description": "The name of the stack to
 create.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to seven tags (key/value
 pairs) for the stack.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 7
```

Nell' StackTemplateId area, si vede che esiste un modello di stack specifico per questo CT e lo schema e il relativo ID è un valore di proprietà obbligatorio.

Lo schema consente di etichettare lo stack che si sta creando, per scopi di contabilità interna. Inoltre, alcune opzioni, come backup, richiedono un tag di tipo key:backup e value:TRUE. Per informazioni approfondite, leggi Tagging Your Amazon EC2 Resources.

```
    },
    "TimeoutInMinutes": {
      "description": "The amount of time, in minutes,
 to allow for creation of the stack.",
      "type": "number",
      "minimum": 0,
      "maximum": 60
    },
    "Parameters": {
      "description": "Specifications for the
 stack.",
      "type": "object",
      "properties": {
        "AccessControl": {
          "description": "The canned (predefined)
 access control list (ACL) to assign to the bucket.",
          "type": "string",
          "enum": [
            "Private",
            "PublicRead",
            "AuthenticatedRead",
            "BucketOwnerRead"
          ]
        },
        "BucketName": {
          "description": "A name for the bucket.
 The bucket name must contain only lowercase letters,
 numbers, periods (.), and hyphens (-).",
          "type": "string",
          "pattern": "^[a-z0-9]([-.a-z0-9]+)[a-z
0-9]$",
          "minLength": 3,
          "maxLength": 63
        }
      },
      "additionalProperties": false,
      "required": [
        "AccessControl",
        "BucketName"
      ]
    }
  },
  "additionalProperties": false,
  "required": [
```

La sezione Parametri dello schema CT JSON è dove fornisci i parametri di esecuzione.

Per questo schema, BucketName sono richiesti solo l'ACL e i parametri di esecuzione.

```
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Richiedi un nuovo CT

Dopo aver esaminato lo schema, potresti decidere che non fornisce parametri sufficienti per creare la distribuzione desiderata. In tal caso, esamina i CloudFormation modelli esistenti per trovarne uno più vicino a quello desiderato. Una volta individuati i parametri aggiuntivi necessari, invia un messaggio Gestione | Altro | Altro | Crea CT.

> **Note**
>
> All Other | Other Create and Update CTs riceve l'attenzione di un operatore AMS, che ti contatterà per discutere del nuovo CT.

Per inviare una richiesta per un nuovo CT, accedi alla console AMS tramite la normale procedura AWS Management Consolee segui questi passaggi.

1. Dalla barra di navigazione a sinistra, fai clic su RFCs.

   Si apre la pagina del RFCs pannello di controllo.

2. Fai clic su Create (Crea).

   Viene visualizzata la pagina Crea una richiesta di modifica.

3. Seleziona Gestione nell'elenco a discesa Categoria e Altro per la sottocategoria e l'articolo. Per l'operazione, scegli Crea. La RFC avrà bisogno dell'approvazione prima di poter essere implementata.

4. Inserisci le informazioni sul motivo per cui desideri il CT, ad esempio: richiesta di un CT di archiviazione Create S3 modificato che consenta la personalizzazione ACLs, in base al CT di archiviazione Create S3 esistente. Ciò dovrebbe comportare un nuovo CT: Deployment |

Advanced Stack Components | S3 storage | Create S3 custom ACL. Questo nuovo CT potrebbe essere pubblico.

5. Fare clic su Submit (Invia).

   Il tuo RFC viene visualizzato sulla dashboard RFC.

Prova il nuovo CT

Una volta che AWS Managed Services ha creato il nuovo CT, lo testerai inviando una RFC. Se hai collaborato con AMS per rendere preapprovata la nuova CT, puoi semplicemente seguire una richiesta RFC standard e controllare il risultato (per i dettagli sull'invio RFCs, consulta Creazione e invio di una RFC). Se il nuovo CT non è preapprovato (vuoi essere sicuro che non venga mai eseguito senza un'approvazione esplicita), dovrai discuterne l'implementazione con AMS ogni volta che vuoi eseguirlo.

Avviamenti rapidi

Argomenti

Utilizzando una combinazione di tipi di modifica AMS, è possibile eseguire attività complesse.

Puoi utilizzare il sistema di gestione delle modifiche AMS per configurare AMS Resource Scheduler, per un account multi-account landing zone (MALZ) o per un account di landing zone a account singolo (SALZ). Il processo varia. Inoltre, per effettuare trasferimenti di file e istantanee tra account.

Avvio rapido di AMS Resource Scheduler

Utilizza questa guida rapida per implementare AMS Resource Scheduler, uno scheduler di istanze basato su tag per ridurre i costi in AMS Advanced.

L'AMS Resource Scheduler si basa su AWS Instance Scheduler.

Terminologia AMS Resource Scheduler

Prima di iniziare, è bene conoscere la terminologia di AMS Resource Scheduler:

- periodo: ogni pianificazione deve contenere almeno un periodo che definisca l'ora o le ore in cui l'istanza deve essere eseguita. Una pianificazione può contenere più di un periodo. Quando in una pianificazione vengono utilizzati più periodi, Resource Scheduler applica l'azione di avvio appropriata quando almeno una delle regole del periodo è vera.
- fuso orario: per un elenco di valori di fuso orario accettabili da utilizzare nel DefaultTimezoneparametro a cui si fa riferimento in seguito, vedere la colonna TZ dell'Elenco dei fusi orari del database TZ.
- ibernazione: se impostata su true, EC2 le istanze abilitate per l'ibernazione e che soddisfano i requisiti di ibernazione vengono ibernate (). suspend-to-disk Controlla la EC2 console per scoprire se le tue istanze sono abilitate per l'ibernazione. Usa l'ibernazione per le EC2 istanze Amazon interrotte che eseguono Amazon Linux.

- enforced: se impostato su true, in base alla pianificazione definita, Resource Scheduler arresta una risorsa in esecuzione se viene avviata manualmente al di fuori del periodo di esecuzione e avvia una risorsa se viene interrotta manualmente durante il periodo di esecuzione.
- retain_running: se impostato su true, impedisce a Resource Scheduler di arrestare un'istanza alla fine di un periodo di esecuzione se l'istanza è stata avviata manualmente prima dell'inizio del periodo. Ad esempio, se un'istanza con un periodo configurato che va dalle 9:00 alle 17:00 viene avviata manualmente prima delle 9:00, Resource Scheduler non interrompe l'istanza alle 17:00.
- ssm-maintenance-window: aggiunge una finestra AWS Systems Manager di manutenzione come periodo di esecuzione a una pianificazione. Quando specifichi il nome di una finestra di manutenzione esistente nello stesso account e nella stessa regione AWS dello stack distribuito per pianificare EC2 le istanze Amazon, Resource Scheduler avvierà l'istanza prima dell'inizio della finestra di manutenzione e la interromperà al termine della finestra di manutenzione, se nessun altro periodo di esecuzione specifica che l'istanza deve essere eseguita e se l'evento di manutenzione è completato.

  Il Resource Scheduler utilizza la AWS Lambda frequenza specificata durante la configurazione iniziale per determinare quanto tempo manca alla finestra di manutenzione per avviare l'istanza. Se si imposta il AWS CloudFormation parametro Frequency su un valore pari o inferiore a 10 minuti, Resource Scheduler avvia l'istanza 10 minuti prima della finestra di manutenzione. Se si imposta la frequenza su un valore superiore a 10 minuti, Resource Scheduler avvia l'istanza per lo stesso numero di minuti della frequenza specificata. Ad esempio, se si imposta la frequenza della finestra di manutenzione di Systems Manager su 30 minuti, Resource Scheduler avvia l'istanza 30 minuti prima della finestra di manutenzione.

  Per ulteriori informazioni, vedere AWS Systems Manager Manutenzione di Windows.
- override-status: sostituisce temporaneamente le azioni di avvio e arresto del Resource Scheduler configurate dal Resource Scheduler. Se si imposta il campo su run, Resource Scheduler avvia, ma non arresta, l'istanza applicabile. L'istanza viene eseguita finché non viene interrotta manualmente. Se si imposta lo stato di sostituzione su Interrotto, il Resource Scheduler si arresta ma non avvia l'istanza applicabile. L'istanza non viene eseguita finché non viene avviata manualmente.

implementazione di AMS Resource Scheduler

Per implementare una soluzione AMS Resource scheduler, segui questi passaggi.

1. Invia una RFC Deployment | AMS Resource Scheduler | Solution | Deploy (ct-0ywnhc8e5k9z5) e fornisci i seguenti parametri:

- SchedulingActive: Sì per abilitare la pianificazione delle risorse, No per disabilitarla. L'impostazione predefinita è Sì.
- ScheduledServices: immettere un elenco di servizi separati da virgole per cui pianificare le risorse. I valori validi includono una combinazione di autoscaling, ec2 e rds. L'impostazione predefinita è autoscaling, ec2, rds.
- TagName: il nome della chiave Tag che associa gli schemi di pianificazione delle risorse alle risorse del servizio. L'impostazione predefinita è Schedule.

  > **Note**
  >
  > La distribuzione di Resource Scheduler funzionerà solo su risorse con questo tag.

- DefaultTimezone: il nome del fuso orario, nel formato US/Pacific, da utilizzare come fuso orario predefinito. L'impostazione predefinita è UTC.

2. Dopo aver ricevuto la conferma che la RFC nel primo passaggio è stata eseguita correttamente, è possibile inviare il tipo Periodo | Aggiungi modifica.
3. Infine, invia una RFC per aggiungere una pianificazione al periodo creato nel secondo passaggio. Usa il tipo di modifica Schedule | Add.

implementazione e utilizzo di AMS Resource Scheduler FAQs

Domande frequenti su AMS Resource Scheduler.

D: Cosa succede se abilito l'ibernazione ma l' EC2 istanza non la supporta?

R: L'ibernazione salva i contenuti dalla memoria dell'istanza (RAM) nel volume root di Amazon Elastic Block Store (Amazon EBS). Se questo campo è impostato su true, le istanze vengono ibernate quando Resource Scheduler le interrompe.

Se si imposta Resource Scheduler per utilizzare l'ibernazione ma le istanze non sono abilitate per l'ibernazione o non soddisfano i prerequisiti di ibernazione, Resource Scheduler registra un avviso e le istanze vengono arrestate senza ibernazione. Per ulteriori informazioni, consulta Hibernate Your Instance.

D: Cosa succede se imposto sia override_status che enforced?

R: Se imposti override_status su running e imposti enforced su true (impedisce che un'istanza venga avviata manualmente al di fuori di un periodo di esecuzione), Resource Scheduler interrompe l'istanza.

Se imposti override_status su stop e imposti enforced su true (impedisce che un'istanza venga arrestata manualmente durante un periodo di esecuzione), Resource Scheduler riavvia l'istanza.

> **Note**
>
> Se enforced è false, viene applicato il comportamento di override configurato.

D: Dopo aver distribuito AMS Resource Scheduler, come posso disabilitare o abilitare il Resource Scheduler nel mio account?

R: Per disabilitare o abilitare AMS Resource Scheduler:

- Per disabilitare: crea un RFC usando State | Disable. Assicurati di impostarlo su SchedulerStateDISABLE
- Per abilitare: crea una RFC usando State | Enable. Assicurati di impostare su SchedulerStateENABLE

D Cosa succede se il periodo di validità di AMS Resource Scheduler rientra nella finestra di manutenzione prevista per l'applicazione delle patch?

R: Resource Scheduler funziona in base alle pianificazioni configurate. Se è configurato per arrestare un'istanza mentre è in corso l'applicazione della patch, interrompe l'istanza a meno che la finestra di applicazione delle patch non venga aggiunta come periodo alla pianificazione prima dell'inizio dell'applicazione delle patch. In altre parole, Resource Scheduler non avvia automaticamente le istanze interrotte per l'applicazione di patch a meno che non sia configurato un periodo designato. Per evitare conflitti con la finestra di manutenzione delle patch, aggiungi la finestra temporale assegnata per l'applicazione delle patch alla pianificazione di Resource Scheduler come periodo. Per aggiungere un periodo alla pianificazione esistente, crea una RFC utilizzando Period | Add.

D Se ho bisogno di una pianificazione diversa per EC2 istanze diverse, posso impostare più di una pianificazione all'interno del mio account?

R: Sì, puoi creare più pianificazioni. Ogni pianificazione può avere più periodi in base al requisito. Quando AMS Resource Scheduler è abilitato nell'account, viene configurata una Tag Key. Ad

esempio, se la chiave tag è «Schedule», il valore del tag può differire in base a diverse pianificazioni, che corrispondono al nome della pianificazione di AMS Resource Scheduler. Per aggiungere una nuova pianificazione, puoi creare una RFC utilizzando il tipo di modifica Management | AMS Resource Scheduler | Schedule | Add (ct-2bxelbn765ive), vedi Schedule | Add.

D: Dove posso trovare tutti i diversi tipi di modifica supportati per AMS Resource Scheduler?

R: AMS dispone dei tipi di modifica di Resource Scheduler per distribuire AMS Resource Scheduler sul tuo account, abilitarlo o disabilitarlo, definire, aggiungere, aggiornare ed eliminare pianificazioni e periodi da utilizzare con esso e descrivere (ottieni una descrizione dettagliata) di pianificazioni e periodi.

Configurazione di backup su più account (intra-regione)

AWS Backup supporta la possibilità di copiare istantanee da un account all'altro all'interno della stessa regione AWS purché i due account si trovino all'interno della stessa AWS Organization. Ad esempio, in AMS Advanced multi-account landing zone (MALZ), puoi configurare una copia dello snapshot su più account all'interno della stessa regione AWS utilizzando questo quick-start.

Per ulteriori informazioni, consulta AWS Backup e AWS Organizations introducono la funzionalità di backup tra account

Le istantanee vengono copiate su più account per il disaster recovery (DR). Potresti avere l'obbligo di conservare le istantanee all'interno della stessa regione AWS, ma oltre i confini dell'account, per la protezione dei dati.

Panoramica:

Ad alto livello, questi sono i passaggi per i backup su più account all'interno di AMS:

- Crea un account di destinazione per ospitare i backup nella regione AWS in cui è ospitata la tua landing zone AMS (fase 1)
- Crea una chiave KMS per crittografare i backup nell'account di destinazione (passaggio 3)
- Crea un vault di backup nell'account di destinazione della stessa regione della tua landing zone AMS Advanced (fase 4)
- Abilita l'impostazione cross-account nel tuo account di gestione (passaggio 5)
- Crea o modifica il piano e le regole di backup dell'account di origine (fase 6)

> **Note**
>
> Assicurati che l'account di origine e quello di destinazione si trovino nella stessa regione. Se desideri copiare i tuoi backup in più regioni, contatta la tua CA o il CSDM.

Per abilitare e configurare i backup su più account:

1. Crea un account di destinazione per ospitare i backup; se disponi già di un account di questo tipo, puoi saltare questo passaggio. Per creare l'account, invia una RFC dal tuo account Management Payer utilizzando il tipo di modifica Deployment | Managed landing zone | Management account | Create application account (con VPC) (ct-1zdasmc2ewzrs).
2. [Facoltativo] Se le risorse o le istantanee sono crittografate nell'account di origine (ad esempio, Prod), condividi la chiave KMS utilizzata per la crittografia con l'account di destinazione. A tale scopo, invia una RFC utilizzando Management | Advanced stack components | KMS key | Update change type (ct-3ovo7px2vsa6n).
3. Nell'account di destinazione, crea una chiave KMS da utilizzare per la crittografia di Backup Vault. Per fare ciò, invia una RFC utilizzando Deployment | Advanced stack components | KMS key | Create (auto) change type (ct-1d84keiri1jhg).
4. Nell'account di destinazione, crea un Backup Vault utilizzando la chiave creata in precedenza. AWS Backup Vaults può essere creato utilizzando il tipo di modifica automatica CFN Ingest, Deployment | Ingestion | Stack from CloudFormation Template | Create (ct-36cn2avfrrj9v). Nella stessa richiesta, è necessario modificare la politica di accesso al vault per consentire agli account di origine di accedere al vault. Ecco un esempio di policy:

   CloudFormation Modello di esempio per un Backup Vault:

```
{
  "Description": "Test infrastructure",
  "Resources": {
  "BackupVaultForTesting": {
    "Type": "AWS::Backup::BackupVault",
    "Properties": {
      "BackupVaultName": "backup-vault-for-test",
      "EncryptionKeyArn" : "arn:aws:kms:us-east-2:123456789012:key/227d8xxx-
aefx-44ex-a09x-b90c487b4xxx",
        "AccessPolicy" : {
          "Version": "2012-10-17",
          "Statement": [
```

```
            {
              "Sid": "AllowSrcAccountPermissionsToCopy",
              "Effect": "Allow",
              "Action": "backup:CopyIntoBackupVault",
              "Resource": "*",
              "Principal": {
                "AWS": ["arn:aws:iam::987654321098:root"]
              }
            }
          ]
        }
      }
    }
  }
}
```

5. Dal tuo account Management Payer, abilita il backup su più account. A tale scopo, invia una RFC utilizzando il tipo di modifica Management | AWS Backup | Backup plan | Enable cross account copy (Management account) (ct-2yja7ihh30ply).
6. Infine, dall'account di origine da cui provengono i backup, crea la regola o le regole del piano di backup che regola i backup per copiare le istantanee su più account. A tale scopo, invia una RFC utilizzando Deployment | AWS Backup | Backup plan | Create change type (ct-2hyozbpa0sx0m). Se devi aggiornare un piano di backup esistente, invia una RFC utilizzando il tipo di modifica Gestione | Altro | Altro | Aggiornamento (ct-0xdawir96cy7k) con queste informazioni:

   1. Il nome del piano di backup e il nome della regola da aggiornare.
   2. L'ARN del vault di backup dell' destination/ICE account.
   3. La conservazione per days/months cui desideri conservare le istantanee nel vault ICE di destinazione.

Tutorial

Argomenti

I seguenti tutorial descrivono in dettaglio i passaggi per creare uno stack a due livelli con High Availability (ct-06mjngx5flwto), utilizzare la CLI e utilizzare la Console e distribuire un gruppo Amazon Auto Scaling (ASG) Linux o RHEL. EC2 Segue un tier-and-tie tutorial simile (uno per la console e uno per la CLI), che utilizza separati CTs, creati in un ordine tale da consentire di collegare insieme le risorse man mano che vengono create.

Le descrizioni di tutte le opzioni CT, incluse, sono ChangeTypeId disponibili nel managedservices/latest/ctref /Change Type Reference.

Tutorial sulla console: Stack a due livelli ad alta disponibilità (Linux/RHEL)

Questa sezione descrive come implementare un WordPress sito ad alta disponibilità (HA) in un ambiente AMS utilizzando la console AMS.

> **Note**
>
> Questa procedura dettagliata di implementazione è stata testata in ambienti AMZN Linux e RHEL.

Riepilogo delle attività e dei requisiti richiesti: RFCs

1. Crea un'infrastruttura (stack HA a due livelli)
2. Crea un bucket S3 per le applicazioni CodeDeploy
3. Crea il pacchetto di WordPress applicazioni e caricalo nel bucket S3
4. Distribuisci l'applicazione con CodeDeploy

5. Accedi al WordPress sito e accedi per convalidare la distribuzione
6. Distruggi la distribuzione

Le descrizioni di tutte le opzioni CT, incluse ChangeTypeId, sono disponibili in AMS Change Type Reference.

Prima di iniziare

The Deployment | Advanced Stack Components | High Availability Two Tier Stack | Create CT crea un gruppo Auto Scaling, un load balancer, un database e CodeDeploy un nome di applicazione e un gruppo di implementazione (con lo stesso nome assegnato all'applicazione). Per informazioni su CodeDeploy , consulta What is? CodeDeploy

Questa procedura dettagliata utilizza uno stack RFC a due livelli ad alta disponibilità che include `UserData` e descrive anche come creare un WordPress pacchetto da distribuire. CodeDeploy

L'esempio `UserData` illustrato nell'esempio ottiene i metadati dell'istanza, come l'ID dell'istanza, la regione e così via, dall'interno di un'istanza in esecuzione interrogando il servizio di metadati dell'istanza disponibile all'indirizzo http://169.254.169.254/latest/meta-data/. EC2 Questa riga dello script dei dati utente:`REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/ | sed 's/[a-z]$//')`, recupera il nome della zona di disponibilità dal servizio di metadati nella variabile $REGION per le nostre regioni supportate e lo utilizza per completare l'URL per il bucket S3 in cui viene scaricato l'agente. CodeDeploy L'IP 169.254.169.254 è instradabile solo all'interno del VPC (tutti possono interrogare il servizio). VPCs Per informazioni sul servizio, consulta Metadati dell'istanza e dati utente. Nota anche che gli script immessi come UserData vengono eseguiti come utente «root» e non è necessario utilizzare il comando «sudo».

Questa procedura dettagliata lascia i seguenti parametri al valore predefinito (mostrato):

- Gruppo Auto Scaling:. `Cooldown=300, DesiredCapacity=2, EBSOptimized=false, HealthCheckGracePeriod=600, IAMInstanceProfile=customer-mc-ec2-instance-profile, InstanceDetailedMonitoring=true, InstanceRootVolumeIops=0, InstanceRootVolumeType=standard, InstanceType=m3.medium, MaxInstances=2, MinInstances=2, ScaleDownPolicyCooldown=300, ScaleDownPolicyEvaluationPeriods=4, ScaleDownPolicyPeriod=60, ScaleDownPolicyScalingAdjustment=-1, ScaleDownPolicyStatistic=Average, ScaleDownPolicyThreshold=35, ScaleMetricName=CPUUtilization, ScaleUpPolicyCooldown=60, ScaleUpPolicyEvaluationPeriods=2,`

`ScaleUpPolicyPeriod=60, ScaleUpPolicyScalingAdjustment=2, ScaleUpPolicyStatistic=Average, ScaleUpPolicyThreshold=75`

- Load Balancer:. `HealthCheckInterval=30, HealthCheckTimeout=5`
- Banca dati:`BackupRetentionPeriod=7, Backups=true, InstanceType=db.m3.medium, IOPS=0, MultiAZ=true, PreferredBackupWindow=22:00-23:00, PreferredMaintenanceWindow=wed:03:32-wed:04:02, StorageEncrypted=false, StorageEncryptionKey="", StorageType=gp2`.
- Applicazione:`DeploymentConfigName=CodeDeployDefault.OneAtATime`.

Parametri variabili:

La console offre un'opzione ASAP per l'ora di inizio e questa procedura dettagliata consiglia di utilizzarla. ASAP fa sì che la RFC venga eseguita non appena vengono approvate.

> **Note**
>
> Ci sono molti parametri che potreste scegliere di impostare in modo diverso da quelli mostrati. I valori per i parametri mostrati nell'esempio sono stati testati ma potrebbero non essere adatti a te. Negli esempi vengono mostrati solo i valori obbligatori. I valori nei *`replaceable`* caratteri devono essere modificati in quanto sono specifici del tuo account.

Crea l'infrastruttura

Questa procedura utilizza lo stack CT a due livelli ad alta disponibilità seguito dallo storage CT Create S3.

La raccolta dei seguenti dati prima di iniziare velocizzerà la distribuzione.

I DATI RICHIESTI HANNO UNO STACK:

- AutoScalingGroup:
  - UserData: Questo valore è fornito in questo tutorial. Include comandi per configurare la risorsa CodeDeploy e avviare l' CodeDeploy agente.
  - AMI-ID: questo valore determina il sistema operativo delle EC2 istanze che verrà avviato dal gruppo Auto Scaling (ASG). Seleziona un'AMI nel tuo account che inizi con «cliente-» e sia del sistema operativo che desideri. Trova AMI IDs nella pagina dei VPCs dettagli della console AMS

VPCs ->. Questa procedura dettagliata è destinata alla ASGs configurazione per l'utilizzo di un'AMI Amazon Linux o RHEL.

- Database:
  - Questi parametri, DBEngineEngineVersion, e LicenseModeldevono essere impostati in base alla situazione, sebbene i valori mostrati nell'esempio siano stati testati. Il tutorial utilizza rispettivamente questi valori:*MySQL*,*8.0.16*,*general-public-license*.
  - Questi parametri, DBNameMasterUserPassword, e MasterUsernamesono necessari per la distribuzione del pacchetto di applicazioni. Il tutorial utilizza rispettivamente questi valori:,*wordpressDB*,*p4ssw0rd*. *admin* Nota che DBName può contenere solo caratteri alfanumerici.
  - Quando inserisci il codice MasterUsernameper il DB RDS, verrà visualizzato in chiaro, quindi accedi al database il prima possibile e modifica la password per garantire la tua sicurezza.
  - Per gli RDSSubnetID, usa due sottoreti private. Inseriscili uno alla volta premendo «Invio» dopo ciascuno. Trova Subnet IDs con il riferimento all'API Per l'AMS SKMS, consulta la scheda Report nella Console AWS Artifact. operation (CLI: list-subnet-summaries) o nella pagina dei dettagli della console AMS -> VPC. VPCs
- LoadBalancer:
  - Imposta questo parametro, Public, su true perché il tutorial utilizza sottoreti ELB pubbliche.
  - ELBSubnetID: utilizza due sottoreti pubbliche. Inseriscili uno alla volta premendo «Invio» dopo ciascuno. Trova Subnet IDs con il riferimento all'API Per l'AMS SKMS, consulta la scheda Report nella Console AWS Artifact. operation (CLI: list-subnet-summaries) o nella pagina dei dettagli della console AMS -> VPC. VPCs
- Applicazione: il ApplicationNamevalore imposta il nome dell'applicazione e il nome del gruppo di distribuzione CodeDeploy . CodeDeploy Lo usi per distribuire la tua applicazione. Deve essere unico nell'account. Per verificare la presenza di CodeDeploy nomi nel tuo account, consulta la CodeDeploy Console. L'esempio utilizza *WordPress* ma, se intendi utilizzare quel valore, assicurati che non sia già in uso.

1. Avvia lo stack ad alta disponibilità.
   a. Nella pagina Crea RFC, seleziona la categoria Deployment, la sottocategoria Standard Stacks, la voce High availability two tier stack e l'operazione Create, dall'elenco.
   b. IMPORTANTE: scegli Advanced e imposta i valori come mostrato.

Devi solo inserire i valori per le opzioni contrassegnate da un asterisco (*), i valori testati sono mostrati nell'esempio; puoi lasciare vuote le opzioni vuote non obbligatorie.

c. Per la sezione Descrizione RFC:

```
Subject: WP-HA-2-Tier-RFC
```

d. Per la sezione Informazioni sulle risorse, impostate i parametri per Database AutoScalingGroupLoadBalancer, Applicazione e Tag.

Inoltre, lo scopo della chiave di tag AppName "" è consentire di cercare facilmente le istanze ASG nella EC2 console; è possibile chiamare questa chiave di tag «Nome» o qualsiasi altro nome di chiave desiderato. Tieni presente che puoi aggiungere fino a 50 tag.

```
UserData:
    #!/bin/bash
    REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/
 | sed 's/[a-z]$//')
    yum -y install ruby httpd
    chkconfig httpd on
    service httpd start
    touch /var/www/html/status
    cd /tmp
    curl -O https://aws-codedeploy-$REGION.s3.amazonaws.com/latest/install
    chmod +x ./install
    ./install auto
    chkconfig codedeploy-agent on
    service codedeploy-agent start
AmiId:              AMI-ID
Description:        WP-HA-2-Tier-Stack

Database:
    LicenseModel:    general-public-license (USE RADIO BUTTON)
    EngineVersion:   8.0.16
    DBEngine:        MySQL
    RDSSubnetIds:    PRIVATE_AZ1 PRIVATE_AZ2 (ENTER ONE AT A TIME PRESSING
 "ENTER" AFTER EACH)
    MasterUserPassword:  p4ssw0rd
    MasterUsername:      admin
    DBName:              wordpressDB

LoadBalancer:
```

```
    Public:              true (USE RADIO BUTTON)
    ELBSubnetIds:        PUBLIC_AZ1 PUBLIC_AZ2

Application:
    ApplicationName:     WordPress

Tags:
    Name:                WP-Rhel-Stack
```

e. Al termine, fai clic su Invia.

2. Accedi al database che hai creato e modifica la password.
3. Avvia uno stack di bucket S3.

   La raccolta dei seguenti dati prima di iniziare velocizzerà la distribuzione.

   BUCKET S3 DI DATI RICHIESTO:

   - VPC-ID: questo valore determina dove si troverà il tuo S3 Bucket. Trova un VPC IDs con il riferimento all'API For the AMS SKMS, consulta la scheda Report nella Console AWS Artifact Operation (list-vpc-summariesCLI:) o nella pagina della console AMS. VPCs
   - BucketName: Questo valore imposta il nome del bucket S3, lo usi per caricare il pacchetto dell'applicazione. Deve essere univoco in tutta l'area dell'account e non può includere lettere maiuscole. L'inclusione dell'ID dell'account come parte di non BucketName è un requisito, ma semplifica l'identificazione del bucket in un secondo momento. Per vedere quali nomi di bucket S3 esistono nell'account, accedi alla console Amazon S3 del tuo account.

   a. Nella pagina Crea RFC, seleziona la categoria Deployment, la sottocategoria Advanced Stack Components, l'elemento S3 storage e l'operazione Create dall'elenco di selezione RFC CT.

   b. Mantieni l'opzione Basic predefinita e imposta i valori come mostrato.

```
Subject:             S3-Bucket-WP-HA-RFC
Description:         S3BucketForWordPressBundles
BucketName:          ACCOUNT_ID-BUCKET_NAME
AccessControl:       Private
VpcId:               VPC_ID
Name:                S3-Bucket-WP-HA-Stack
TimeoutInMinutes:    60
```

c. Al termine, fate clic su Invia. Il bucket distribuito con questo tipo di modifica consente read/write l'accesso completo all'intero account.

Crea, carica e distribuisci l'applicazione

Innanzitutto, create un pacchetto di WordPress applicazioni, quindi utilizzate il CodeDeploy CTs per creare e distribuire l'applicazione.

1. Scarica WordPress, estrai i file e crea un file. cartella /scripts.

   Comando Linux:

   ```
   wget https://github.com/WordPress/WordPress/archive/master.zip
   ```

   Windows: incolla `https://github.com/WordPress/WordPress/archive/master.zip` in una finestra del browser e scarica il file zip.

   Crea una directory temporanea in cui assemblare il pacchetto.

   Linux:

   ```
   mkdir /tmp/WordPress
   ```

   Windows: crea una cartella "WordPress", utilizzerai il percorso della directory in seguito.

2. Estrai il WordPress codice sorgente nella cartella WordPress "" e crea un file. cartella /scripts.

   Linux:

   ```
   unzip master.zip -d /tmp/WordPress_Temp
   cp -paf /tmp/WordPress_Temp/WordPress-master/* /tmp/WordPress
   rm -rf /tmp/WordPress_Temp
   rm -f master
   cd /tmp/WordPress
   mkdir scripts
   ```

   Windows: vai alla cartella "WordPress" che hai creato e lì crea una cartella «scripts».

   Se utilizzate un ambiente Windows, assicuratevi di impostare il tipo di interruzione per i file di script su Unix (LF). In Notepad ++, questa è un'opzione in basso a destra della finestra.

3. Crea il file CodeDeploy appspec.yml, nella WordPress directory (se copi l'esempio, controlla l'indentazione, ogni spazio conta). IMPORTANTE: assicurati che il percorso «sorgente» sia corretto per copiare WordPress i file (in questo caso, nella tua WordPress directory) nella destinazione prevista (/). var/www/html/WordPress Nell'esempio, il file appspec.yml si trova nella directory con WordPress i file, quindi è necessario solo «/». Inoltre, anche se hai usato un'AMI RHEL per il tuo gruppo Auto Scaling, lascia la riga «os: linux» così com'è. Esempio di file appspec.yml:

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/html/WordPress
hooks:
  BeforeInstall:
    - location: scripts/install_dependencies.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/config_wordpress.sh
      timeout: 300
      runas: root
  ApplicationStart:
    - location: scripts/start_server.sh
      timeout: 300
      runas: root
  ApplicationStop:
    - location: scripts/stop_server.sh
      timeout: 300
      runas: root
```

4. Crea script di file bash in. WordPress cartella /scripts.

   Innanzitutto, crea `config_wordpress.sh` con il seguente contenuto (se preferisci, puoi modificare direttamente il file wp-config.php).

   > **Note**
   >
   > Sostituisci *`DBName`* con il valore fornito nell'HA Stack RFC (ad esempio,`wordpress`). Sostituisci *`DB_MasterUsername`* con il `MasterUsername` valore fornito nell'HA Stack RFC (ad esempio,). `admin`

Sostituisci *DB_MasterUserPassword* con il `MasterUserPassword` valore fornito nell'HA Stack RFC (ad esempio,). `p4ssw0rd`
Sostituire *DB_ENDPOINT* con il nome DNS dell'endpoint negli output di esecuzione di HA Stack RFC (ad esempio,). `srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com` Puoi trovarlo con l'GetRfcoperazione (CLI: get-rfc --rfc-id RFC_ID) o nella pagina dei dettagli RFC della console AMS per l'HA Stack RFC che hai inviato in precedenza.

```
#!/bin/bash
chmod -R 755 /var/www/html/WordPress
cp /var/www/html/WordPress/wp-config-sample.php /var/www/html/WordPress/wp-config.php
cd /var/www/html/WordPress
sed -i "s/database_name_here/DBName/g" wp-config.php
sed -i "s/username_here/DB_MasterUsername/g" wp-config.php
sed -i "s/password_here/DB_MasterUserPassword/g" wp-config.php
sed -i "s/localhost/DB_ENDPOINT/g" wp-config.php
```

5. `install_dependencies.sh`Nella stessa directory, crea con il seguente contenuto:

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS viene installato come parte dei dati utente al momento del lancio per consentire il funzionamento dei controlli sanitari sin dall'inizio.

6. Nella stessa directory, crea `start_server.sh` con il seguente contenuto:

   - Per le istanze Amazon Linux, usa questo:

```
#!/bin/bash
service httpd start
```

- Per le istanze RHEL, usa questo (i comandi aggiuntivi sono politiche che consentono a SELINUX di accettare): WordPress

```
#!/bin/bash
setsebool -P  httpd_can_network_connect_db 1
setsebool -P  httpd_can_network_connect 1
chcon -t httpd_sys_rw_content_t /var/www/html/WordPress/wp-content -R
restorecon -Rv /var/www/html
service httpd start
```

7. Nella stessa directory create `stop_server.sh` con il seguente contenuto:

```
#!/bin/bash
service httpd stop
```

8. Crea il pacchetto zip.

   Linux:

```
$ cd /tmp/WordPress
$ zip -r wordpress.zip .
```

   Windows: vai alla cartella WordPress "" e seleziona tutti i file e crea un file zip, assicurati di chiamarlo wordpress.zip.

1. Carica il pacchetto dell'applicazione nel bucket S3

   Il pacchetto deve essere attivo per continuare a distribuire lo stack.

   Hai automaticamente accesso a qualsiasi istanza di bucket S3 che crei. Puoi accedervi tramite i tuoi Bastions (vedi Accesso alle istanze) o tramite la console S3 e caricare il CodeDeploy pacchetto con drag-and-drop, oppure sfogliando e selezionando il file.

   Puoi anche usare il seguente comando in una finestra di shell; assicurati di avere il percorso corretto del file zip:

```
aws s3 cp wordpress/wordpress.zip s3://BUCKET_NAME/
```

2. Distribuite l' WordPress CodeDeploy Application Bundle

CODICE DI DATI RICHIESTO, DISTRIBUZIONE DELL'APPLICAZIONE:

- CodeDeployApplicationName: Il nome che hai assegnato all' CodeDeploy applicazione.
- CodeDeployGroupName: Poiché l' CodeDeploy applicazione e il gruppo sono stati entrambi creati dal nome assegnato all' CodeDeploy applicazione nello stack HA RFC, questo nome è lo stesso del. CodeDeployApplicationName
- S3Bucket: il nome che hai assegnato al bucket S3.
- S3 BundleType e S3Key: fanno parte del pacchetto di applicazioni che hai distribuito. WordPress
- VpcId: Il VPC pertinente.

a. Nella pagina Crea RFC, seleziona la categoria Distribuzione, la sottocategoria Applicazioni, CodeDeploy l'elemento applicazione e l'operazione Deploy dall'elenco di selezione RFC CT.

b. Mantieni l'opzione Basic predefinita e imposta i valori come mostrato.

> **Note**
>
> Fai riferimento all' CodeDeploy applicazione, al gruppo CodeDeploy di distribuzione, al bucket S3 e al bundle creati in precedenza.

```
Subject:                                WP-CD-Deploy-RFC
Description:                            DeployWordPress
S3Bucket:                               BUCKET_NAME
S3Key:                                  wordpress.zip
S3BundleType:                           zip
CodeDeployApplicationName:              WordPress
CodeDeployDeploymentGroupName:          WordPress
CodeDeployIgnoreApplicationStopFailures: false
RevisionType:                           S3



VpcId:                                  VPC_ID
Name:                                   WP-CD-Deploy-Op
TimeoutInMinutes:                       60
```

c. Al termine, fai clic su Invia.

Convalida della distribuzione dell'applicazione

Passa all'endpoint (LoadBalancerCName) del sistema di bilanciamento del carico creato in precedenza, con il percorso distribuito:/. WordPress WordPress Ad esempio:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Dovresti vedere una pagina come questa:

Eliminare l'implementazione ad alta disponibilità

Per interrompere l'implementazione, invii il Delete Stack CT sullo stack HA Two-Tier e il bucket S3, e puoi richiedere l'eliminazione delle istantanee RDS (vengono eliminate automaticamente dopo dieci giorni, ma costano una piccola somma). Raccogli lo stack IDs per lo stack HA e il bucket S3, quindi segui questi passaggi. Vedi Stack | Elimina.

Tutorial sulla console: implementazione di un sito Web Tier and Tie WordPress

Questa sezione descrive come implementare un WordPress sito ad alta disponibilità (HA) in un ambiente AMS utilizzando la console AMS. Questo set di istruzioni include un esempio di creazione del necessario file di WordPress CodeDeploy pacchetto compatibile (ad esempio zip). L'approvvigionamento delle risorse segue un ordine che consente di collegarle insieme per formare «livelli».

> **Note**
>
> Questa procedura dettagliata di implementazione è progettata per l'uso con un sistema operativo Linux AMZN.
> I parametri delle variabili essenziali sono indicati come*replaceable*; tuttavia, è possibile modificare altri parametri in base alla propria situazione.

Riepilogo delle attività e delle attività richieste RFCs:

1. Crea l'infrastruttura:

   a. Creare un cluster di database MySQL RDS
   b. Creazione di un sistema di bilanciamento del carico
   c. Crea un gruppo Auto Scaling e collegalo al load balancer
   d. Crea un bucket S3 per le applicazioni CodeDeploy
2. Crea un pacchetto di WordPress applicazioni (non richiede una RFC)
3. Distribuisci il pacchetto di WordPress applicazioni con: CodeDeploy
   a. Crea un'applicazione CodeDeploy
   b. Creare un gruppo CodeDeploy di distribuzione
   c. Carica il tuo pacchetto di WordPress applicazioni nel bucket S3 (non richiede una RFC)
   d. Implementa l' CodeDeploy applicazione
4. Convalida la distribuzione
5. Distruggi la distribuzione

Le descrizioni di tutte le opzioni CT, incluse ChangeTypeId , sono disponibili in AMS Change Type Reference.

Creazione di un RFC utilizzando la console (nozioni di base)

Questi sono alcuni passaggi da seguire ogni volta che si crea una RFC utilizzando la console.

1. Fai clic RFCsnel riquadro di navigazione a sinistra per aprire la pagina dell' RFCs elenco, quindi fai clic su Crea RFC.

   Viene visualizzata la pagina Crea RFC.
2. Scegli Sfoglia i tipi di modifica (impostazione predefinita) o Scegli per categoria.
3. Sfoglia i tipi di modifica:

   a. Fai clic su un'opzione di creazione rapida per iniziare una RFC con uno dei tipi di modifica più utilizzati.

      Si apre l'area di configurazione generale per quel tipo di modifica, la riga dell'oggetto viene compilata. Per visualizzare i dettagli del tipo di modifica, apri l'area nella parte superiore della pagina.

   b. Utilizza l'area Tutti i tipi di modifica.

Filtra, passa da una visualizzazione a schede a una tabella o ordina i tipi di modifica. Quando trovi quello che desideri, selezionalo e fai clic su Crea RFC nella parte superiore della pagina.

Si apre l'area di configurazione generale per quel tipo di modifica, la riga dell'oggetto viene compilata. Per visualizzare i dettagli del tipo di modifica, apri l'area nella parte superiore della pagina.

4. Scegli per categoria:

   a. Seleziona la categoria, la sottocategoria, l'articolo e l'operazione appropriati.

      La casella dei dettagli del tipo di modifica viene visualizzata nella parte inferiore della pagina.

   b. Fai clic su Crea RFC nella parte inferiore della pagina.

   c. Si apre l'area di configurazione generale per quel tipo di modifica, viene compilata la riga dell'oggetto. Per visualizzare i dettagli del tipo di modifica, apri l'area nella parte superiore della pagina.

5. Per garantire che determinate persone ricevano notifiche sull'avanzamento della RFC, inserisci gli indirizzi e-mail. Per aggiungere dettagli sul tipo di modifica, compila la Descrizione. Apri l'area di configurazione aggiuntiva per aggiungere ulteriori dettagli sulla RFC.
6. Per Pianificazione, seleziona Esegui questa modifica al più presto o Pianifica questa modifica. Se selezioni Esegui questa modifica il prima possibile, la tua RFC verrà eseguita non appena le approvazioni sono state superate. Se si seleziona Pianifica questo tipo di modifica, vengono visualizzati un calendario, un'ora e un fuso orario selezionati e la RFC viene avviata, dopo l'invio, come pianificato.
7. Nell'area di configurazione dell'esecuzione, configura i parametri del tipo di modifica. Per visualizzare i parametri opzionali, aprite l'area di configurazione aggiuntiva.
8. Quando sei pronto, fai clic su Esegui.

Creazione dell'infrastruttura

Accedi alla console AWS per l'account AMS di destinazione e quindi alla console AMS per l'account.

Le procedure seguenti descrivono la creazione di un database RDS, un load balancer e un gruppo Auto Scaling in modo tale da utilizzare la IDs risorsa per creare l'infrastruttura.

Crea uno stack RDS

Vedi Stack RDS | Create.

Crea uno stack ELB

Lanciare un ELB pubblico.

DATI RICHIESTI:

- `VpcId`: Il VPC che stai utilizzando, dovrebbe essere lo stesso del VPC usato in precedenza.
- `ELBSubnetIds`: Una serie di sottoreti su cui il load balancer distribuirà il traffico. Scegli tra sottoreti pubbliche o private. Trova Subnet IDs con il riferimento all'API Per l'AMS SKMS, consulta la scheda Report nella Console AWS Artifact. operation (CLI: list-subnet-summaries) o nella pagina dei dettagli della console AMS -> VPC. VPCs
- `VpcId`: Il VPC che stai utilizzando, dovrebbe essere lo stesso del VPC usato in precedenza.

1. Nella pagina Crea RFC, seleziona la categoria Deployment, la sottocategoria Advanced Stack Components, la voce Load balancer (ELB) stack e fai clic su Crea. Scegliete Advanced e accettate tutte le impostazioni predefinite (incluse quelle senza valore) ad eccezione di quelle mostrate di seguito.

```
Subject:                        WP-ELB-RFC
ELBSubnetIds:                   PUBLIC_AZ1
                              PUBLIC_AZ2
ELBScheme                       true
ELBCookieExpirationPeriod       600
VpcId:                          VPC_ID
Name:                           WP-Public-ELB
```

2. Al termine, fate clic su Invia.

Crea uno stack di gruppi con Auto Scaling

Avvia un gruppo di Auto Scaling.

DATI RICHIESTI:

- `VpcId`: Il VPC che stai utilizzando, dovrebbe essere lo stesso del VPC usato in precedenza.

- `AMI-ID`: Questo valore determina il tipo di EC2 istanze che verrà avviato dal gruppo Auto Scaling (ASG). Assicurati di selezionare nel tuo account un'AMI che inizi con «cliente-» e sia del sistema operativo che desideri. Trova AMI IDs con il riferimento all'API Per l'AMS SKMS, consulta la scheda Report nella Console AWS Artifact. operation (CLI: list-amis) o nella pagina dei dettagli della console AMS ->. VPCs VPCs Questa procedura dettagliata è destinata alla ASGs configurazione per l'utilizzo di un'AMI Linux.
- `ASGLoadBalancerNames`: Il sistema di bilanciamento del carico che hai creato in precedenza: trova il nome guardando EC2 Console -> Load Balancers (nel menu di navigazione a sinistra). Nota che questo non è il «Nome» che hai specificato quando hai creato l'ELB in precedenza.

1. Nella pagina Crea RFC, seleziona la categoria Deployment, la sottocategoria Advanced Stack Components, la voce Auto scaling group e fai clic su Crea. Scegliete Avanzate e accettate tutte le impostazioni predefinite (incluse quelle senza valore) ad eccezione di quelle mostrate di seguito.

   > **Note**
   >
   > Specificare l'AMI AMS più recente. Specificate l'ELB creato in precedenza.

```
Subject:                          WP-ASG-RFC
ASGSubnetIds:                     PRIVATE_AZ1
                                                  PRIVATE_AZ2
ASGAmiId:                         AMI_ID
VpcId:                            VPC_ID
Name:                             WP_ASG
ASGLoadBalancerNames:             ELB_NAME
ASGUserData:
#!/bin/bash
REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/ | sed
 's/[a-z]$//')
yum -y install ruby httpd
chkconfig httpd on
service httpd start
touch /var/www/html/status
cd /tmp
curl -O https://aws-codedeploy-$REGION.s3.amazonaws.com/latest/install
chmod +x ./install
```

```
./install auto
chkconfig codedeploy-agent on
service codedeploy-agent start
```

2. Al termine, fate clic su Invia.

Crea uno stack S3

Avvia un bucket S3. Nel bucket S3 puoi caricare il pacchetto di applicazioni che hai creato.

DATI RICHIESTI:

- `VPC-ID`: Questo valore determina dove sarà il tuo bucket S3, dovrebbe essere lo stesso del VPC usato in precedenza.
- `AccessControl`: Le opzioni della AccessControl lista preimpostata (ACL) sono, e. `Private` `PublicRead` Per ulteriori informazioni, consulta Amazon Simple Storage Service Canned ACL.
- `BucketName`: Questo valore imposta il nome del bucket S3, lo usi per caricare il pacchetto dell'applicazione. Deve essere univoco in tutta l'area dell'account e non può includere lettere maiuscole. L'inclusione dell'ID dell'account come parte di non BucketName è un requisito, ma semplifica l'identificazione del bucket in un secondo momento. Per vedere quali nomi di bucket S3 esistono nell'account, accedi alla console Amazon S3 del tuo account.

1. Nella pagina Crea RFC, seleziona la categoria Deployment, la sottocategoria Advanced Stack Components, la voce Storage S3 e fai clic su Crea.

   È possibile lasciare l'opzione del parametro predefinito su Basic per accettare i valori predefiniti come descritto. Per impostare valori diversi, scegliete Avanzato.

   > **Note**
   >
   > Il bucket distribuito con questo tipo di modifica consente read/write l'accesso completo all'intero account, potrebbero essere necessari nuovi tipi di modifica per consentire autorizzazioni di accesso più limitate.

   ```
   Subject:            S3-Bucket-RFC
   BucketName:         ACCOUNT_ID-codedeploy-bundles
   AccessControl:      Private
   ```

```
VpcId:                  VPC_ID
Name:                   S3BucketForWP
```

2. Al termine, fai clic su Invia.

Crea un WordPress CodeDeploy pacchetto

La sezione fornisce un esempio di creazione di un pacchetto di distribuzione delle applicazioni.

1. Scarica WordPress, estrai i file e crea un file. cartella /scripts.

   Comando Linux:

   ```
   wget https://github.com/WordPress/WordPress/archive/master.zip
   ```

   Windows: incolla `https://github.com/WordPress/WordPress/archive/master.zip` in una finestra del browser e scarica il file zip.

   Crea una directory temporanea in cui assemblare il pacchetto.

   Linux:

   ```
   mkdir /tmp/WordPress
   ```

   Windows: crea una cartella "WordPress", utilizzerai il percorso della directory in seguito.

2. Estrai il WordPress codice sorgente nella cartella WordPress "" e crea un file. cartella /scripts.

   Linux:

   ```
   unzip master.zip -d /tmp/WordPress_Temp
   cp -paf /tmp/WordPress_Temp/WordPress-master/* /tmp/WordPress
   rm -rf /tmp/WordPress_Temp
   rm -f master
   cd /tmp/WordPress
   mkdir scripts
   ```

   Windows: vai alla cartella "WordPress" che hai creato e lì crea una cartella «scripts».

Se utilizzate un ambiente Windows, assicuratevi di impostare il tipo di interruzione per i file di script su Unix (LF). In Notepad ++, questa è un'opzione in basso a destra della finestra.

3. Crea il file CodeDeploy appspec.yml, nella WordPress directory (se copi l'esempio, controlla l'indentazione, ogni spazio conta). IMPORTANTE: assicurati che il percorso «sorgente» sia corretto per copiare WordPress i file (in questo caso, nella tua WordPress directory) nella destinazione prevista (/). var/www/html/WordPress Nell'esempio, il file appspec.yml si trova nella directory con WordPress i file, quindi è necessario solo «/». Inoltre, anche se hai usato un'AMI RHEL per il tuo gruppo Auto Scaling, lascia la riga «os: linux» così com'è. Esempio di file appspec.yml:

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/html/WordPress
hooks:
  BeforeInstall:
    - location: scripts/install_dependencies.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/config_wordpress.sh
      timeout: 300
      runas: root
  ApplicationStart:
    - location: scripts/start_server.sh
      timeout: 300
      runas: root
  ApplicationStop:
    - location: scripts/stop_server.sh
      timeout: 300
      runas: root
```

4. Crea script di file bash in. WordPress cartella /scripts.

   Innanzitutto, crea `config_wordpress.sh` con il seguente contenuto (se preferisci, puoi modificare direttamente il file wp-config.php).

> **Note**
>
> Sostituisci *DBName* con il valore fornito nell'HA Stack RFC (ad esempio,`wordpress`).
> Sostituisci *DB_MasterUsername* con il `MasterUsername` valore fornito nell'HA Stack RFC (ad esempio,). `admin`
> Sostituisci *DB_MasterUserPassword* con il `MasterUserPassword` valore fornito nell'HA Stack RFC (ad esempio,). `p4ssw0rd`
> Sostituire *DB_ENDPOINT* con il nome DNS dell'endpoint negli output di esecuzione di HA Stack RFC (ad esempio,). `srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com` Puoi trovarlo con l'GetRfcoperazione (CLI: get-rfc --rfc-id RFC_ID) o nella pagina dei dettagli RFC della console AMS per l'HA Stack RFC che hai inviato in precedenza.

```
#!/bin/bash
chmod -R 755 /var/www/html/WordPress
cp /var/www/html/WordPress/wp-config-sample.php /var/www/html/WordPress/wp-
config.php
cd /var/www/html/WordPress
sed -i "s/database_name_here/DBName/g" wp-config.php
sed -i "s/username_here/DB_MasterUsername/g" wp-config.php
sed -i "s/password_here/DB_MasterUserPassword/g" wp-config.php
sed -i "s/localhost/DB_ENDPOINT/g" wp-config.php
```

5. `install_dependencies.sh`Nella stessa directory, crea con il seguente contenuto:

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS viene installato come parte dei dati utente al momento del lancio per consentire il funzionamento dei controlli sanitari sin dall'inizio.

6. Nella stessa directory, crea `start_server.sh` con il seguente contenuto:

- Per le istanze Amazon Linux, usa questo:

```
#!/bin/bash
service httpd start
```

- Per le istanze RHEL, usa questo (i comandi aggiuntivi sono politiche che consentono a SELINUX di accettare): WordPress

```
#!/bin/bash
setsebool -P  httpd_can_network_connect_db 1
setsebool -P  httpd_can_network_connect 1
chcon -t httpd_sys_rw_content_t /var/www/html/WordPress/wp-content -R
restorecon -Rv /var/www/html
service httpd start
```

7. Nella stessa directory create `stop_server.sh` con il seguente contenuto:

```
#!/bin/bash
service httpd stop
```

8. Crea il pacchetto zip.

   Linux:

```
$ cd /tmp/WordPress
$ zip -r wordpress.zip .
```

   Windows: vai alla cartella WordPress "" e seleziona tutti i file e crea un file zip, assicurati di chiamarlo wordpress.zip.

Distribuisci il pacchetto di WordPress applicazioni con CodeDeploy

CodeDeploy È un servizio di distribuzione AWS che automatizza le distribuzioni di applicazioni su istanze Amazon. EC2 Questa parte del processo prevede la creazione di un' CodeDeploy applicazione, la creazione di un gruppo di CodeDeploy distribuzione e quindi la distribuzione dell'applicazione utilizzando. CodeDeploy

Crea un' CodeDeploy applicazione

L' CodeDeploy applicazione è semplicemente un nome o un contenitore utilizzato da AWS CodeDeploy per garantire che durante una distribuzione venga fatto riferimento alla revisione, alla configurazione di distribuzione e al gruppo di distribuzione corretti. La configurazione di distribuzione, in questo caso, è il WordPress pacchetto creato in precedenza.

DATI RICHIESTI:

- `VpcId`: Il VPC che stai usando dovrebbe essere lo stesso del VPC usato in precedenza.
- `CodeDeployApplicationName`: Deve essere unico nell'account. Controlla la CodeDeploy console per verificare i nomi delle applicazioni esistenti.

1. Crea l' CodeDeploy applicazione per WordPress

   Nella pagina Crea RFC, seleziona la categoria Distribuzione, sottocategoria Applicazioni, elemento CodeDeploy applicazione e operazione Crea dall'elenco di selezione RFC CT. Scegliete Basic e impostate i valori come mostrato. Al termine, fate clic su Invia.

   ```
   Subject:                    CD-WP-App-RFC
   CodeDeployApplicationName:  WordPress
   VpcId:                      VPC_ID
   Name:                       WP-CD-App
   ```

2. Al termine, fai clic su Invia.

Creare un gruppo CodeDeploy di distribuzione

Crea il gruppo CodeDeploy di distribuzione.

Un gruppo CodeDeploy di distribuzione definisce un insieme di istanze individuali destinate a una distribuzione.

DATI RICHIESTI:

- `VpcId`: Il VPC che stai usando dovrebbe essere lo stesso del VPC usato in precedenza.
- `CodeDeployApplicationName`: Usa il valore che hai creato in precedenza.
- `CodeDeployAutoScalingGroups`: utilizza il nome del gruppo Auto Scaling creato in precedenza.

- `CodeDeployDeploymentGroupName`: un nome per il gruppo di distribuzione. Questo nome deve essere univoco per ogni applicazione associata al gruppo di distribuzione.
- `CodeDeployServiceRoleArn`: Usa la formula riportata nell'esempio.

1. Nella pagina Crea RFC, selezionare la categoria Distribuzione, la sottocategoria Applicazioni, il gruppo di CodeDeploy distribuzione degli articoli e l'operazione Crea dall'elenco di selezione RFC CT. Scegliete Advanced e impostate i valori come mostrato (per la RFC è necessario solo un oggetto). Al termine, fai clic su Invia.

> **Note**
>
> Fate riferimento all'ARN del ruolo di CodeDeploy servizio in questo formato `"arn:aws:iam::085398962942:role/aws-codedeploy-role"` e utilizzate il nome del gruppo Auto scaling creato in precedenza per «ASG_NAME».

```
Description:                        Create CodeDeploy Deployment Group for WP
CodeDeployApplicationName:          WordPress
CodeDeployAutoScalingGroups:        ASG_NAME
CodeDeployDeploymentConfigName:     CodeDeployDefault.HalfAtATime
CodeDeployDeploymentGroupName:      WP CD Group
CodeDeployServiceRoleArn:           arn:aws:iam::ACCOUNT_ID:role/aws-codedeploy-role

VpcId:                              VPC_ID
Name:                               WP Deployment Group
```

2. Al termine, fate clic su Invia.

Carica l' WordPress applicazione

Hai automaticamente accesso a qualsiasi istanza di bucket S3 che crei. Puoi accedervi tramite i tuoi Bastions (vedi Accesso alle istanze) o tramite la console S3 e caricare il pacchetto. CodeDeploy Il pacchetto deve essere disponibile per continuare a distribuire lo stack. L'esempio utilizza il nome del bucket creato in precedenza.

Puoi usare questo comando AWS per comprimere il pacchetto:

```
aws s3 cp wordpress/wordpress.zip s3://ACCOUNT_ID-codedeploy-bundles/
```

Implementa l' WordPress applicazione con CodeDeploy

Distribuisci l' CodeDeploy applicazione.

DATI RICHIESTI:

- `VPC-ID`: Il VPC che stai utilizzando, dovrebbe essere lo stesso del VPC usato in precedenza.
- `CodeDeployApplicationName`: Usa il nome dell' CodeDeploy applicazione che hai creato in precedenza.
- `CodeDeployDeploymentGroupName`: utilizza il nome del gruppo di CodeDeploy distribuzione creato in precedenza.
- `S3Location`(dove hai caricato il pacchetto di applicazioni):`S3Bucket`: BucketName Quello che hai creato in precedenza `S3BundleType` e`S3Key`: il tipo e il nome del pacchetto che hai inserito nel tuo negozio S3.

1. Distribuisci l'Application Bundle WordPress CodeDeploy

   Nella pagina Crea RFC, seleziona la categoria Distribuzione, la sottocategoria Applicazioni, CodeDeploy l'elemento applicazione e l'operazione Deploy dall'elenco di selezione di RFC CT. Scegliete Basic e impostate i valori come mostrato. Al termine, fate clic su Invia.

   > **Note**
   >
   > Fai riferimento all' CodeDeploy applicazione, al gruppo CodeDeploy di distribuzione, al bucket S3 e al bundle creati in precedenza.

   ```
   Subject:                          WP-CD-Deploy-RFC
   CodeDeployApplicationName:        WordPress
   CodeDeployDeploymentGroupName:    WPCDGroup
   RevisionType:                     S3
   S3Bucket:                         ACCOUNT_ID-codedeploy-bundles
   S3BundleType:                     zip
   S3Key:                            wordpress.zip
   VpcId:                            VPC_ID
   Name:                             WordPress
   ```

2. Al termine, fai clic su Invia.

Convalida della distribuzione dell'applicazione

Passa all'endpoint (ELB CName) del sistema di bilanciamento del carico creato in precedenza, con il percorso distribuito:/. WordPress WordPress Ad esempio:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Abbatti la distribuzione delle applicazioni

Per interrompere la distribuzione, inviate il Delete Stack CT allo stack di database RDS, all'application load balancer, al gruppo Auto Scaling, al bucket S3 e all'applicazione e al gruppo Code Deploy, sei in tutto. RFCs Inoltre, puoi inviare una richiesta di servizio per l'eliminazione degli snapshot RDS (vengono eliminati automaticamente dopo dieci giorni, ma costano una piccola somma). Raccogli lo stack IDs per tutti e poi segui questi passaggi. Vedi Stack | Elimina.

Tutorial CLI: Stack a due livelli ad alta disponibilità (Linux/RHEL)

Questa sezione descrive come implementare uno stack a due livelli ad alta disponibilità (HA) in un ambiente AMS utilizzando l'AMS CLI.

> **Note**
>
> Questa procedura dettagliata di implementazione è stata testata in ambienti AMZN Linux e RHEL.

Riepilogo delle attività e dei requisiti richiesti: RFCs

1. Crea un'infrastruttura (stack HA a due livelli)
2. Crea un bucket S3 per le applicazioni CodeDeploy
3. Crea il pacchetto di WordPress applicazioni e caricalo nel bucket S3
4. Distribuisci l'applicazione con CodeDeploy
5. Accedi al WordPress sito e accedi per convalidare la distribuzione

Prima di iniziare

The Deployment | Advanced Stack Components | High Availability Two Tier Stack Advanced | Create CT crea un gruppo Auto Scaling, un load balancer, un database e CodeDeploy un nome di applicazione e un gruppo di implementazione (con lo stesso nome assegnato all'applicazione). Per informazioni su CodeDeploy , consulta What is? CodeDeploy

Questa procedura dettagliata utilizza un RFC High Availability Two-Tier Stack (Advanced) che include UserData e descrive anche come creare un WordPress pacchetto da distribuire. CodeDeploy

L'esempio `UserData` illustrato nell'esempio ottiene i metadati dell'istanza, come l'ID dell'istanza, la regione e così via, dall'interno di un'istanza in esecuzione interrogando il servizio di metadati dell'istanza disponibile all'indirizzo http://169.254.169.254/latest/meta-data/. EC2 Questa riga dello script dei dati utente:`REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/ | sed 's/[a-z]$//')`, recupera il nome della zona di disponibilità dal servizio di metadati nella variabile $REGION per le nostre regioni supportate e lo utilizza per completare l'URL per il bucket S3 in cui viene scaricato l'agente. CodeDeploy L'IP 169.254.169.254 è instradabile solo all'interno del VPC (tutti possono interrogare il servizio). VPCs Per informazioni sul servizio, consulta Metadati dell'istanza e dati utente. Nota anche che gli script immessi come UserData vengono eseguiti come utente «root» e non è necessario utilizzare il comando «sudo».

Questa procedura dettagliata lascia i seguenti parametri al valore predefinito (mostrato):

- Gruppo Auto Scaling:. `Cooldown=300, DesiredCapacity=2, EBSOptimized=false, HealthCheckGracePeriod=600, IAMInstanceProfile=customer-mc-ec2-instance-profile, InstanceDetailedMonitoring=true, InstanceRootVolumeIops=0, InstanceRootVolumeType=standard, InstanceType=m3.medium, MaxInstances=2, MinInstances=2, ScaleDownPolicyCooldown=300, ScaleDownPolicyEvaluationPeriods=4, ScaleDownPolicyPeriod=60, ScaleDownPolicyScalingAdjustment=-1, ScaleDownPolicyStatistic=Average, ScaleDownPolicyThreshold=35, ScaleMetricName=CPUUtilization, ScaleUpPolicyCooldown=60, ScaleUpPolicyEvaluationPeriods=2, ScaleUpPolicyPeriod=60, ScaleUpPolicyScalingAdjustment=2, ScaleUpPolicyStatistic=Average, ScaleUpPolicyThreshold=75`
- Load Balancer:. `HealthCheckInterval=30, HealthCheckTimeout=5`
- Banca dati:`BackupRetentionPeriod=7, Backups=true, InstanceType=db.m3.medium, IOPS=0, MultiAZ=true, PreferredBackupWindow=22:00-23:00,`

`PreferredMaintenanceWindow=wed:03:32-wed:04:02, StorageEncrypted=false, StorageEncryptionKey="", StorageType=gp2`.

- Applicazione:`DeploymentConfigName=CodeDeployDefault.OneAtATime`.
- Secchio S3:. `AccessControl=Private`

IMPOSTAZIONI AGGIUNTIVE:

`RequestedStartTimee RequestedEndTime` se desideri pianificare la tua RFC: puoi usare Time.is per determinare l'ora UTC corretta. Gli esempi forniti devono essere modificati in modo appropriato. Una RFC non può procedere se è trascorsa l'ora di inizio. In alternativa, puoi lasciare questi valori disattivati per creare un RFC ASAP che venga eseguito non appena vengono approvate.

> **Note**
>
> Ci sono molti parametri che potreste scegliere di impostare in modo diverso da quelli mostrati. I valori dei parametri mostrati nell'esempio sono stati testati ma potrebbero non essere adatti a te.

Crea l'infrastruttura

La raccolta dei seguenti dati prima di iniziare velocizzerà la distribuzione.

I DATI RICHIESTI HANNO UNO STACK:

- AutoScalingGroup:
  - `UserData`: Questo valore è fornito in questo tutorial. Include comandi per configurare la risorsa CodeDeploy e avviare l' CodeDeploy agente.
  - `AMI-ID`: Questo valore determina il tipo di EC2 istanze che verrà avviato dal gruppo Auto Scaling (ASG). Assicurati di selezionare nel tuo account un'AMI che inizi con «cliente-» e sia del sistema operativo che desideri. Trova AMI IDs con il riferimento all'API Per l'AMS SKMS, consulta la scheda Report nella Console AWS Artifact. operation (CLI: list-amis) o nella pagina dei dettagli della console AMS ->. VPCs VPCs Questa procedura dettagliata è destinata alla ASGs configurazione per l'utilizzo di un'AMI Linux.
- Database:
  - Questi parametri, `DBEngineEngineVersion`, `LicenseModel` devono essere impostati in base alla situazione, sebbene i valori mostrati nell'esempio siano stati testati.

  - Questi parametri,`RDSSubnetIds`, `DBNameMasterUsername`, e `MasterUserPassword` sono necessari per la distribuzione del pacchetto di applicazioni. Per gli RDSSubnet ID, utilizzate due sottoreti private.
- LoadBalancer:
  - Questi parametri, `DBEngineEngineVersion`, `LicenseModel` devono essere impostati in base alla situazione, sebbene i valori mostrati nell'esempio siano stati testati.
  - `ELBSubnetIds`: utilizza due sottoreti pubbliche.
- Applicazione: il `ApplicationName` valore imposta il nome dell' CodeDeploy applicazione e il nome del gruppo CodeDeploy di distribuzione. Lo usi per distribuire la tua applicazione. Deve essere unico nell'account. Per verificare la presenza di CodeDeploy nomi nel tuo account, consulta la CodeDeploy Console. L'esempio utilizza "WordPress" ma, se intendi utilizzare quel valore, assicurati che non sia già in uso.

Questa procedura utilizza lo stack a due livelli ad alta disponibilità (avanzato) CT (ct-06mjngx5flwto) e lo storage CT Create S3 (ct-1a68ck03fn98r). Dal tuo account autenticato, segui questi passaggi nella riga di comando.

1. Avvia lo stack di infrastruttura.

   a. Invia lo schema JSON dei parametri di esecuzione per lo stack CT a due livelli HA in un file nella cartella corrente denominata .json. CreateStackParams

   ```
   aws amscm get-change-type-version --change-type-id "ct-06mjngx5flwto"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateStackParams.json
   ```

   b. Modifica lo schema. Sostituire *`variables`* il file appropriato. Ad esempio, utilizzate il sistema operativo desiderato per EC2 le istanze che l'ASG creerà. Registratele `ApplicationName` mentre le utilizzerete in seguito per distribuire l'applicazione. Tieni presente che puoi aggiungere fino a 50 tag.

   ```
   {
   "Description":      "HA two tier stack for WordPress",
   "Name":             "WordPressStack",
   "TimeoutInMinutes":  360,
   "Tags": [
           {
               "Key": "ApplicationName",
   ```

```
            "Value": "WordPress"
        }
    ],
"AutoScalingGroup": {
            "AmiId":    "AMI-ID",
            "UserData": "#!/bin/bash \n
            REGION=$(curl 169.254.169.254/latest/meta-data/placement/
availability-zone/ | sed 's/[a-z]$//') \n
            yum -y install ruby httpd \n
            chkconfig httpd on \n
            service httpd start \n
            touch /var/www/html/status \n
            cd /tmp \n
            curl -O https://aws-codedeploy-$REGION.s3.amazonaws.com/latest/
install \n
            chmod +x ./install \n
            ./install auto \n
            chkconfig codedeploy-agent on \n
            service codedeploy-agent start"
    },
    "LoadBalancer": {
        "Public":               true,
        "HealthCheckTarget":    "HTTP:80/status"
    },
    "Database":     {
        "DBEngine":             "MySQL",
        "DBName":               "wordpress",
        "EngineVersion":        "8.0.16 ",
        "LicenseModel":         "general-public-license",
        "MasterUsername":       "admin",
        "MasterUserPassword":   "p4ssw0rd"
    },
    "Application":  {
    "ApplicationName":  "WordPress"
        }
}
```

c. Esporta il modello CreateRfc JSON in un file nella cartella corrente denominata CreateStackRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateStackRfc.json
```

d. Modifica il modello RFC come segue e salvalo, puoi eliminare e sostituire i contenuti. Tieni presente che `RequestedStartTime` ora `RequestedEndTime` sono facoltativi; la loro esclusione crea un RFC ASAP che viene eseguito non appena viene approvato (cosa che di solito avviene automaticamente). Per inviare una RFC pianificata, aggiungi questi valori.

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-06mjngx5flwto",
"Title":                "HA-Stack-For-WP-RFC"
}
```

e. Crea la RFC, specificando il CreateStackRfc file.json e il file dei parametri di esecuzione .json: CreateStackParams

```
aws amscm create-rfc --cli-input-json file://CreateStackRfc.json --execution-
parameters file://CreateStackParams.json
```

Riceverai l'ID RFC nella risposta. Salva l'ID per i passaggi successivi.

f. Invia la RFC:

```
aws amscm submit-rfc --rfc-id  RFC_ID
```

Se la RFC ha esito positivo, non riceverai alcun output.

g. Per verificare lo stato della RFC, esegui

```
aws amscm get-rfc --rfc-id RFC_ID
```

Prendi nota dell'ID RFC.

2. Avvia un bucket S3

La raccolta dei seguenti dati prima di iniziare velocizzerà la distribuzione.

BUCKET S3 DI DATI RICHIESTO:

- `VPC-ID`: questo valore determina dove sarà il tuo S3 Bucket. Usa lo stesso ID VPC che hai usato in precedenza.

- `BucketName`: Questo valore imposta il nome del bucket S3, lo usi per caricare il pacchetto dell'applicazione. Deve essere univoco in tutta l'area dell'account e non può includere lettere maiuscole. L'inclusione dell'ID dell'account come parte di non BucketName è un requisito, ma semplifica l'identificazione del bucket in un secondo momento. Per vedere quali nomi di bucket S3 esistono nell'account, accedi alla console Amazon S3 del tuo account.

a. Invia lo schema JSON dei parametri di esecuzione per lo storage S3, crea CT in un file JSON denominato CreateS3 .json. StoreParams

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateS3StoreParams.json
```

b. Modifica lo schema come segue, puoi eliminare e sostituire i contenuti. Sostituisci *VPC_ID* in modo appropriato. I valori dell'esempio sono stati testati, ma potrebbero non essere adatti a te.

> **Tip**
>
> `BucketName`Devono essere univoci in tutta l'area dell'account e non possono includere lettere maiuscole. L'inclusione dell'ID dell'account come parte di non BucketName è un requisito, ma semplifica l'identificazione del bucket in un secondo momento. Per vedere quali nomi di bucket S3 esistono nell'account, accedi alla console Amazon S3 del tuo account.

```
{
"Description":      "S3BucketForWordPressBundle",
"VpcId":            "VPC_ID",
"StackTemplateId":  "stm-s2b72beb000000000",
"Name":             "S3BucketForWP",
"TimeoutInMinutes":  60,
"Parameters":   {
    "AccessControl":    "Private",
    "BucketName":       "ACCOUNT_ID-BUCKET_NAME"
    }
}
```

c. Esporta il modello JSON in un file, nella cartella corrente, denominato CreateS3 .json: CreateRfc StoreRfc

```
aws amscm create-rfc --generate-cli-skeleton > CreateS3StoreRfc.json
```

d. Modifica e salva il file CreateS3 StoreRfc .json, puoi eliminare e sostituire il contenuto. Tieni presente che `RequestedStartTime` ora `RequestedEndTime` sono facoltativi; la loro esclusione crea un RFC ASAP che viene eseguito non appena viene approvato (cosa che di solito avviene automaticamente). Per inviare una RFC pianificata, aggiungi questi valori.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1a68ck03fn98r",
"Title":                "S3-Stack-For-WP-RFC"
}
```

e. Crea la RFC, specificando il file CreateS3 .json e il file dei parametri di esecuzione CreateS3 StoreRfc .json: StoreParams

```
aws amscm create-rfc --cli-input-json file://CreateS3StoreRfc.json  --
execution-parameters file://CreateS3StoreParams.json
```

Riceverai il nuovo RFC nella risposta. RfcId Salva l'ID per i passaggi successivi.

f. Invia la RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Se la RFC ha esito positivo, non riceverai alcun output.

g. Per verificare lo stato della RFC, esegui

```
aws amscm get-rfc --rfc-id RFC_ID
```

Crea, carica e distribuisci l'applicazione

Innanzitutto, create un pacchetto di WordPress applicazioni, quindi utilizzate il CodeDeploy CTs per creare e distribuire l'applicazione.

1. Scarica WordPress, estrai i file e crea un file. cartella /scripts.

Comando Linux:

```
wget https://github.com/WordPress/WordPress/archive/master.zip
```

Windows: incolla `https://github.com/WordPress/WordPress/archive/master.zip` in una finestra del browser e scarica il file zip.

Crea una directory temporanea in cui assemblare il pacchetto.

Linux:

```
mkdir /tmp/WordPress
```

Windows: crea una cartella "WordPress", utilizzerai il percorso della directory in seguito.

2. Estrai il WordPress codice sorgente nella cartella WordPress "" e crea un file. cartella /scripts.

   Linux:

   ```
   unzip master.zip -d /tmp/WordPress_Temp
   cp -paf /tmp/WordPress_Temp/WordPress-master/* /tmp/WordPress
   rm -rf /tmp/WordPress_Temp
   rm -f master
   cd /tmp/WordPress
   mkdir scripts
   ```

   Windows: vai alla cartella "WordPress" che hai creato e lì crea una cartella «scripts».

   Se utilizzate un ambiente Windows, assicuratevi di impostare il tipo di interruzione per i file di script su Unix (LF). In Notepad ++, questa è un'opzione in basso a destra della finestra.

3. Crea il file CodeDeploy appspec.yml, nella WordPress directory (se copi l'esempio, controlla l'indentazione, ogni spazio conta). IMPORTANTE: assicurati che il percorso «sorgente» sia corretto per copiare WordPress i file (in questo caso, nella tua WordPress directory) nella destinazione prevista (/). var/www/html/WordPress Nell'esempio, il file appspec.yml si trova nella directory con WordPress i file, quindi è necessario solo «/». Inoltre, anche se hai usato un'AMI RHEL per il tuo gruppo Auto Scaling, lascia la riga «os: linux» così com'è. Esempio di file appspec.yml:

   ```
   version: 0.0
   ```

```
os: linux
files:
  - source: /
    destination: /var/www/html/WordPress
hooks:
  BeforeInstall:
    - location: scripts/install_dependencies.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/config_wordpress.sh
      timeout: 300
      runas: root
  ApplicationStart:
    - location: scripts/start_server.sh
      timeout: 300
      runas: root
  ApplicationStop:
    - location: scripts/stop_server.sh
      timeout: 300
      runas: root
```

4. Crea script di file bash in. WordPress cartella /scripts.

   Innanzitutto, crea `config_wordpress.sh` con il seguente contenuto (se preferisci, puoi modificare direttamente il file wp-config.php).

   > **Note**
   >
   > Sostituisci *`DBName`* con il valore fornito nell'HA Stack RFC (ad esempio,`wordpress`).
   > Sostituisci *`DB_MasterUsername`* con il `MasterUsername` valore fornito nell'HA Stack RFC (ad esempio,). `admin`
   > Sostituisci *`DB_MasterUserPassword`* con il `MasterUserPassword` valore fornito nell'HA Stack RFC (ad esempio,). `p4ssw0rd`
   > Sostituire *`DB_ENDPOINT`* con il nome DNS dell'endpoint negli output di esecuzione di HA Stack RFC (ad esempio,). `srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com` Puoi trovarlo con l'GetRfcoperazione (CLI: get-rfc --rfc-id RFC_ID) o nella pagina dei dettagli RFC della console AMS per l'HA Stack RFC che hai inviato in precedenza.

```
#!/bin/bash
chmod -R 755 /var/www/html/WordPress
cp /var/www/html/WordPress/wp-config-sample.php /var/www/html/WordPress/wp-
config.php
cd /var/www/html/WordPress
sed -i "s/database_name_here/DBName/g" wp-config.php
sed -i "s/username_here/DB_MasterUsername/g" wp-config.php
sed -i "s/password_here/DB_MasterUserPassword/g" wp-config.php
sed -i "s/localhost/DB_ENDPOINT/g" wp-config.php
```

5. `install_dependencies.sh`Nella stessa directory, crea con il seguente contenuto:

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS viene installato come parte dei dati utente al momento del lancio per consentire il funzionamento dei controlli sanitari sin dall'inizio.

6. Nella stessa directory, crea `start_server.sh` con il seguente contenuto:

   - Per le istanze Amazon Linux, usa questo:

   ```
   #!/bin/bash
   service httpd start
   ```

   - Per le istanze RHEL, usa questo (i comandi aggiuntivi sono politiche che consentono a SELINUX di accettare): WordPress

   ```
   #!/bin/bash
   setsebool -P  httpd_can_network_connect_db 1
   setsebool -P  httpd_can_network_connect 1
   chcon -t httpd_sys_rw_content_t /var/www/html/WordPress/wp-content -R
   restorecon -Rv /var/www/html
   service httpd start
   ```

7. Nella stessa directory create `stop_server.sh` con il seguente contenuto:

```
#!/bin/bash
service httpd stop
```

8. Crea il pacchetto zip.

   Linux:

```
$ cd /tmp/WordPress
$ zip -r wordpress.zip .
```

   Windows: vai alla cartella WordPress "" e seleziona tutti i file e crea un file zip, assicurati di chiamarlo wordpress.zip.

1. Carica il pacchetto dell'applicazione nel bucket S3.

   Il pacchetto deve essere disponibile per continuare a distribuire lo stack.

   Hai automaticamente accesso a qualsiasi istanza di bucket S3 che crei. Puoi accedervi tramite i tuoi bastioni o tramite la console S3 e caricare il WordPress pacchetto drag-and-drop o sfogliare e selezionare il file zip.

   Puoi anche usare il seguente comando in una finestra di shell; assicurati di avere il percorso corretto del file zip:

```
aws s3 cp wordpress.zip s3://BUCKET_NAME/
```

2. Distribuite il pacchetto di WordPress applicazioni.

   La raccolta dei seguenti dati prima di iniziare velocizzerà la distribuzione.

   DATI RICHIESTI:

   - `VPC-ID`: Questo valore determina dove si troverà il tuo S3 Bucket. Usa lo stesso ID VPC che hai usato in precedenza.
   - `CodeDeployApplicationNameeCodeDeployApplicationName`: il `ApplicationName` valore utilizzato nell'RFC HA 2-Tier Stack imposta e il. CodeDeployApplicationName CodeDeployDeploymentGroupName L'esempio utilizza "WordPress" ma potresti aver usato un valore diverso.

- `S3Location`: Per`S3Bucket`, usa quello `BucketName` che hai creato in precedenza. I `S3BundleType` e `S3Key` provengono dal pacchetto che hai inserito nel tuo negozio S3.

a. Emetti lo schema JSON dei parametri di esecuzione per l' CodeDeploy applicazione, distribuisci CT in un file JSON denominato Deploy Params.json. CDApp

```
aws amscm get-change-type-version --change-type-id "ct-2edc3sd1sqmrb"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 DeployCDAppParams.json
```

b. Modifica lo schema come segue e salvalo come, puoi eliminare e sostituire i contenuti.

```
{
"Description":                          "DeployWPCDApp",
"VpcId":                                "VPC_ID",
"Name":                                 "WordPressCDAppDeploy",
"TimeoutInMinutes":                     60,
"Parameters":   {
    "CodeDeployApplicationName":                "WordPress",
    "CodeDeployDeploymentGroupName":            "WordPress",
    "CodeDeployIgnoreApplicationStopFailures":   false,
    "CodeDeployRevision": {
      "RevisionType": "S3",
      "S3Location": {
        "S3Bucket":     "BUCKET_NAME",
        "S3BundleType": "zip",
        "S3Key":        "wordpress.zip" }
        }
    }
}
```

c. Esporta il modello JSON CreateRfc per in un file, nella cartella corrente, denominato Deploy CDApp RFC.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeployCDAppRfc.json
```

d. Modifica e salva il file Deploy CDApp RFC.json, puoi eliminare e sostituire il contenuto. Tieni presente che `RequestedStartTime` ora `RequestedEndTime` sono facoltativi; la loro esclusione crea un RFC ASAP che viene eseguito non appena viene approvato (cosa che di solito avviene automaticamente). Per inviare una RFC pianificata, aggiungi questi valori.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2edc3sd1sqmrb",
"Title":                "CD-Deploy-For-WP-RFC"
}
```

e. Crea la RFC, specificando il file Deploy CDApp Rfc e il file dei parametri di esecuzione CDApp Deploy Params:

```
aws amscm create-rfc --cli-input-json file://DeployCDAppRfc.json  --execution-
parameters file://DeployCDAppParams.json
```

Riceverai il nuovo RFC nella RfcId risposta. Salva l'ID per i passaggi successivi.

f. Invia la RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Se la RFC ha esito positivo, non riceverai alcun output.

g. Per verificare lo stato della RFC, esegui

```
aws amscm get-rfc --rfc-id RFC_ID
```

Convalida della distribuzione dell'applicazione

Passa all'endpoint (ELB CName) del sistema di bilanciamento del carico creato in precedenza, con il percorso distribuito:/. WordPress WordPress Ad esempio:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Riduci la distribuzione delle applicazioni

Una volta terminato il tutorial, ti consigliamo di interrompere l'implementazione in modo da non farti pagare per le risorse.

Di seguito è riportata un'operazione generica di eliminazione dello stack. Ti consigliamo di inviarla due volte, una per lo stack HA 2-Tier e una volta per lo stack di bucket S3. Come ultima operazione, invia una richiesta di servizio per eliminare tutte le istantanee per il bucket S3 (include l'ID dello stack

del bucket S3 nella richiesta di servizio). Vengono eliminati automaticamente dopo 10 giorni, ma eliminarli anticipatamente consente di risparmiare un po' di costi.

Questa procedura dettagliata fornisce un esempio di utilizzo della console AMS per eliminare uno stack S3; questa procedura si applica all'eliminazione di qualsiasi stack utilizzando la console AMS.

> **Note**
>
> Se si elimina un bucket S3, è necessario prima svuotarlo degli oggetti.

DATI RICHIESTI:

- `StackId`: Lo stack da usare. Puoi trovarlo consultando la pagina AMS Console Stacks, disponibile tramite un link nel menu di navigazione a sinistra. Utilizzando l'API/CLI AMS SKMS, esegui il riferimento per l'API AMS SKMS, consulta la scheda Report nella Console AWS Artifact. Operazione (nella CLI). `list-stack-summaries`
- L'ID del tipo di modifica per questa procedura dettagliata è «1.0"`ct-0q0bic0ywqk6c`, per scoprire la versione più recente, esegui questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-0q0bic0ywqk6c
```

CREAZIONE IN LINEA:

- Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea). E

```
aws amscm create-rfc --change-type-id "ct-0q0bic0ywqk6c" --change-type-version "1.0"
 --title "Delete My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

- Invia la RFC utilizzando l'ID RFC restituito nell'operazione di creazione RFC. Fino all'invio, la RFC rimane nello `Editing` stato e non viene applicata alcuna modifica.

```
aws amscm submit-rfc --rfc-id RFC_ID
```

- Monitora lo stato RFC e visualizza l'output di esecuzione:

```
aws amscm get-rfc --rfc-id RFC_ID
```

CREAZIONE DEL MODELLO:

1. Esporta il modello RFC in un file nella cartella corrente; l'esempio lo chiama DeleteStackRfc .json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteStackRfc.json
   ```

2. Modifica e salva il file.json. DeleteStackRfc Poiché l'eliminazione di uno stack ha un solo parametro di esecuzione, i parametri di esecuzione possono essere contenuti nel DeleteStackRfc file.json stesso (non è necessario creare un file JSON separato con parametri di esecuzione).

   Le virgolette interne dell'estensione ExecutionParameters JSON devono essere eliminate con una barra rovesciata (\). Esempio senza ora di inizio e fine:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0q0bic0ywqk6c",
   "Title":                "Delete-My-Stack-RFC"
   "ExecutionParameters":  "{
           \"StackId\":\"STACK_ID\"}"
   }
   ```

3. Crea la RFC:

   ```
   aws amscm create-rfc --cli-input-json file://DeleteStackRfc.json
   ```

   Riceverai RfcId il nuovo RFC nella risposta. Ad esempio:

   ```
   {
   "RfcId": "daaa1867-ffc5-1473-192a-842f6b326102"
   }
   ```

   Salva l'ID per i passaggi successivi.

4. Invia la RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Se la RFC ha esito positivo, non riceverai alcuna conferma dalla riga di comando.

5. Per monitorare lo stato della richiesta e visualizzare l'output di esecuzione:

```
aws amscm get-rfc --rfc-id RFC_ID --query "Rfc.
{Status:Status.Name,Exec:ExecutionOutput}" --output table
```

Tutorial CLI: implementazione di un sito Web Tier and Tie WordPress

Questa sezione descrive come implementare un WordPress sito ad alta disponibilità (HA) in un ambiente AMS utilizzando l'AMS CLI. Questo set di istruzioni include un esempio di creazione del necessario file di pacchetto WordPress CodeDeploy compatibile (ad esempio zip).

> **Note**
>
> Questa procedura dettagliata di implementazione è progettata per l'uso con un ambiente Linux AMZN.
> I parametri delle variabili essenziali sono indicati come*replaceable*; tuttavia, è possibile modificare altri parametri in base alla propria situazione.

Riepilogo delle attività e delle attività richieste RFCs:

1. Crea l'infrastruttura:
   a. Creare uno stack RDS (CLI)
   b. Creazione di un sistema di bilanciamento del carico
   c. Crea un gruppo di Auto Scaling e collegalo al load balancer
   d. Crea un bucket S3 per le applicazioni CodeDeploy
2. Crea un pacchetto di WordPress applicazioni (non richiede una RFC)
3. Distribuisci il pacchetto di WordPress applicazioni con: CodeDeploy
   a. Crea un'applicazione CodeDeploy
   b. Creare un gruppo CodeDeploy di distribuzione

c. Carica il tuo pacchetto di WordPress applicazioni nel bucket S3 (non richiede una RFC)
d. Implementa l' CodeDeploy applicazione
4. Convalida la distribuzione
5. Distruggi la distribuzione

Segui tutti i passaggi indicati nella riga di comando dal tuo account autenticato.

Creazione di un RFC utilizzando la CLI

Per informazioni dettagliate sulla creazione RFCs, vedere Creazione RFCs; per una spiegazione dei parametri RFC comuni, vedere Parametri comuni RFC.

Crea l'infrastruttura

Le procedure seguenti descrivono la creazione di un database RDS, un load balancer e un gruppo Auto Scaling in modo tale da utilizzare la IDs risorsa per creare l'infrastruttura.

Creare uno stack RDS (CLI)

Vedi RDS stack | Create.

Crea uno stack ELB

Avvia un sistema di bilanciamento del carico pubblico (ELB). Vedi Load Balancer (ELB) Stack | Create.

Crea uno stack di gruppi con Auto Scaling

Avvia un gruppo di Auto Scaling.

Vedi Auto Scaling Group | Create.

Crea un negozio S3

Avvia un bucket S3. Nel bucket S3 puoi caricare il pacchetto di applicazioni che hai creato. Vedi S3 Storage | Create.

Crea un pacchetto di WordPress applicazioni per CodeDeploy

Questa sezione fornisce un esempio di creazione di un pacchetto di distribuzione delle applicazioni.

1. Scarica WordPress, estrai i file e crea un file. cartella /scripts.

   Comando Linux:

   ```
   wget https://github.com/WordPress/WordPress/archive/master.zip
   ```

   Windows: incolla `https://github.com/WordPress/WordPress/archive/master.zip` in una finestra del browser e scarica il file zip.

   Crea una directory temporanea in cui assemblare il pacchetto.

   Linux:

   ```
   mkdir /tmp/WordPress
   ```

   Windows: crea una cartella "WordPress", utilizzerai il percorso della directory in seguito.

2. Estrai il WordPress codice sorgente nella cartella WordPress "" e crea un file. cartella /scripts.

   Linux:

   ```
   unzip master.zip -d /tmp/WordPress_Temp
   cp -paf /tmp/WordPress_Temp/WordPress-master/* /tmp/WordPress
   rm -rf /tmp/WordPress_Temp
   rm -f master
   cd /tmp/WordPress
   mkdir scripts
   ```

   Windows: vai alla cartella "WordPress" che hai creato e lì crea una cartella «scripts».

   Se utilizzate un ambiente Windows, assicuratevi di impostare il tipo di interruzione per i file di script su Unix (LF). In Notepad ++, questa è un'opzione in basso a destra della finestra.

3. Crea il file CodeDeploy appspec.yml, nella WordPress directory (se copi l'esempio, controlla l'indentazione, ogni spazio conta). IMPORTANTE: assicurati che il percorso «sorgente» sia corretto per copiare WordPress i file (in questo caso, nella tua WordPress directory) nella destinazione prevista (/). var/www/html/WordPress Nell'esempio, il file appspec.yml si trova nella directory con WordPress i file, quindi è necessario solo «/». Inoltre, anche se hai usato un'AMI RHEL per il tuo gruppo Auto Scaling, lascia la riga «os: linux» così com'è. Esempio di file appspec.yml:

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/html/WordPress
hooks:
  BeforeInstall:
    - location: scripts/install_dependencies.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/config_wordpress.sh
      timeout: 300
      runas: root
  ApplicationStart:
    - location: scripts/start_server.sh
      timeout: 300
      runas: root
  ApplicationStop:
    - location: scripts/stop_server.sh
      timeout: 300
      runas: root
```

4. Crea script di file bash in. WordPress cartella /scripts.

   Innanzitutto, crea `config_wordpress.sh` con il seguente contenuto (se preferisci, puoi modificare direttamente il file wp-config.php).

   > **Note**
   >
   > Sostituisci *`DBName`* con il valore fornito nell'HA Stack RFC (ad esempio,`wordpress`).
   > Sostituisci *`DB_MasterUsername`* con il `MasterUsername` valore fornito nell'HA Stack RFC (ad esempio,). `admin`
   > Sostituisci *`DB_MasterUserPassword`* con il `MasterUserPassword` valore fornito nell'HA Stack RFC (ad esempio,). `p4ssw0rd`
   > Sostituire *`DB_ENDPOINT`* con il nome DNS dell'endpoint negli output di esecuzione di HA Stack RFC (ad esempio,). `srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com` Puoi trovarlo con l'GetRfcoperazione (CLI: get-rfc --rfc-

id RFC_ID) o nella pagina dei dettagli RFC della console AMS per l'HA Stack RFC che hai inviato in precedenza.

```
#!/bin/bash
chmod -R 755 /var/www/html/WordPress
cp /var/www/html/WordPress/wp-config-sample.php /var/www/html/WordPress/wp-
config.php
cd /var/www/html/WordPress
sed -i "s/database_name_here/DBName/g" wp-config.php
sed -i "s/username_here/DB_MasterUsername/g" wp-config.php
sed -i "s/password_here/DB_MasterUserPassword/g" wp-config.php
sed -i "s/localhost/DB_ENDPOINT/g" wp-config.php
```

5. `install_dependencies.sh`Nella stessa directory, crea con il seguente contenuto:

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS viene installato come parte dei dati utente al momento del lancio per consentire il funzionamento dei controlli sanitari sin dall'inizio.

6. Nella stessa directory, crea `start_server.sh` con il seguente contenuto:

   - Per le istanze Amazon Linux, usa questo:

     ```
     #!/bin/bash
     service httpd start
     ```

   - Per le istanze RHEL, usa questo (i comandi aggiuntivi sono politiche che consentono a SELINUX di accettare): WordPress

     ```
     #!/bin/bash
     setsebool -P  httpd_can_network_connect_db 1
     setsebool -P  httpd_can_network_connect 1
     ```

```
chcon -t httpd_sys_rw_content_t /var/www/html/WordPress/wp-content -R
restorecon -Rv /var/www/html
service httpd start
```

7. Nella stessa directory create `stop_server.sh` con il seguente contenuto:

```
#!/bin/bash
service httpd stop
```

8. Crea il pacchetto zip.

   Linux:

```
$ cd /tmp/WordPress
$ zip -r wordpress.zip .
```

   Windows: vai alla cartella WordPress "" e seleziona tutti i file e crea un file zip, assicurati di chiamarlo wordpress.zip.

Distribuisci l' WordPress Application Bundle con CodeDeploy

CodeDeploy È un servizio di distribuzione AWS che automatizza le distribuzioni di applicazioni su istanze Amazon. EC2 Questa parte del processo prevede la creazione di un' CodeDeploy applicazione, la creazione di un gruppo di CodeDeploy distribuzione e quindi la distribuzione dell'applicazione utilizzando. CodeDeploy

Creare un'applicazione CodeDeploy

L' CodeDeploy applicazione è semplicemente un nome o un contenitore utilizzato da AWS CodeDeploy per garantire che durante una distribuzione venga fatto riferimento alla revisione, alla configurazione di distribuzione e al gruppo di distribuzione corretti. La configurazione di distribuzione, in questo caso, è il WordPress pacchetto creato in precedenza.

DATI RICHIESTI:

- `VpcId`: Il VPC che stai usando dovrebbe essere lo stesso del VPC usato in precedenza.
- `CodeDeployApplicationName`: Deve essere unico nell'account. Controlla la CodeDeploy console per verificare i nomi delle applicazioni esistenti.

- `ChangeTypeIdeChangeTypeVersion`: L'ID del tipo di modifica per questa procedura dettagliata è`ct-0ah3gwb9seqk2`, per scoprire la versione più recente, esegui questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-0ah3gwb9seqk2
```

1. Invia lo schema JSON dei parametri di esecuzione per l' CodeDeploy applicazione CT in un file nella cartella corrente; l'esempio lo chiama Create Params.json. CDApp

```
aws amscm get-change-type-version --change-type-id "ct-0ah3gwb9seqk2" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateCDAppParams.json
```

2. Modificate e salvate il file JSON come segue; potete eliminare e sostituire il contenuto.

```
{
"Description":                      "Create WordPress CodeDeploy App",
"VpcId":                            "VPC_ID",
"StackTemplateId":                  "stm-sft6rv00000000000",
"Name":                             "WordPressCDApp",
"TimeoutInMinutes":                 60,
"Parameters":   {
    "CodeDeployApplicationName":    "WordPressCDApp"
    }
}
```

3. Esporta il modello JSON CreateRfc per in un file nella cartella corrente; l'esempio lo chiama CDApp Create RFC.json.

```
aws amscm create-rfc --generate-cli-skeleton > CreateCDAppRfc.json
```

4. Modifica e salva il file JSON come segue; puoi eliminare e sostituire il contenuto. Nota che `RequestedStartTime` ora `RequestedEndTime` sono facoltativi; la loro esclusione fa sì che la RFC venga eseguita non appena viene approvata (cosa che di solito avviene automaticamente). Per inviare una RFC «pianificata», aggiungi questi valori.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0ah3gwb9seqk2",
"Title":                "CD-App-For-WP-Stack-RFC"
```

```
}
```

5. Crea la RFC, specificando il file Create CDApp Rfc e il file dei parametri di esecuzione:

```
aws amscm create-rfc --cli-input-json file://CreateCDAppRfc.json --execution-
parameters file://CreateCDAppParams.json
```

   Nella risposta riceverai l'ID RFC del nuovo RFC. Salva l'ID per i passaggi successivi.

6. Invia la RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

   Se la RFC ha esito positivo, non riceverai alcun output.

7. Invia la RFC:

```
aws amscm get-rfc --rfc-id RFC_ID
```

Crea un gruppo di CodeDeploy distribuzione

Crea il gruppo CodeDeploy di distribuzione.

Un gruppo CodeDeploy di distribuzione definisce un insieme di istanze individuali destinate a una distribuzione.

DATI RICHIESTI:

- `VpcId`: Il VPC che stai usando dovrebbe essere lo stesso del VPC usato in precedenza.
- `CodeDeployApplicationName`: Usa il valore che hai creato in precedenza.
- `CodeDeployAutoScalingGroups`: utilizza il nome del gruppo Auto Scaling creato in precedenza.
- `CodeDeployDeploymentGroupName`: nome per il gruppo di distribuzione. Questo nome deve essere univoco per ogni applicazione associata al gruppo di distribuzione.
- `CodeDeployServiceRoleArn`: Usa la formula fornita nell'esempio.
- `ChangeTypeIdeChangeTypeVersion`: L'ID del tipo di modifica per questa procedura dettagliata `èct-2gd0u847qd9d2`, per scoprire la versione più recente, esegui questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-2gd0u847qd9d2
```

1. Invia lo schema JSON dei parametri di esecuzione in un file nella cartella corrente; l'esempio lo chiama Create .json. CDDep GroupParams

```
aws amscm get-change-type-version --change-type-id "ct-2gd0u847qd9d2"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateCDDepGroupParams.json
```

2. Modifica e salva il file JSON come segue; puoi eliminare e sostituire il contenuto.

```
{
"Description":                          "CreateWPCDDeploymentGroup",
"VpcId":                                "VPC_ID",
"StackTemplateId":                      "stm-sp9lrk00000000000",
"Name":                                 "WordPressCDAppGroup",
"TimeoutInMinutes":                     60,
"Parameters":   {
    "CodeDeployApplicationName":        "WordPressCDApp",
    "CodeDeployAutoScalingGroups":      ["ASG_NAME"],
    "CodeDeployDeploymentConfigName":   "CodeDeployDefault.HalfAtATime",
    "CodeDeployDeploymentGroupName":    "UNIQUE_CDDepGroupNAME",
    "CodeDeployServiceRoleArn":         "arn:aws:iam::ACCOUNT_ID:role/aws-
codedeploy-role"
    }
}
```

3. Esporta il modello JSON CreateRfc per in un file nella cartella corrente; l'esempio lo chiama CDDep GroupRfc Create .json.

```
aws amscm create-rfc --generate-cli-skeleton > CreateCDDepGroupRfc.json
```

4. Modifica e salva il file JSON come segue; puoi eliminare e sostituire il contenuto. Nota che `RequestedStartTime` ora `RequestedEndTime` sono facoltativi; la loro esclusione fa sì che la RFC venga eseguita non appena viene approvata (cosa che di solito avviene automaticamente). Per inviare una RFC «programmata», aggiungi questi valori.

```
{
```

```
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2gd0u847qd9d2",
"Title":                "CD-Dep-Group-For-WP-Stack-RFC"
}
```

5. Create la RFC, specificando il CDDep GroupRfc file Create e il file dei parametri di esecuzione:

```
aws amscm create-rfc --cli-input-json file://CreateCDDepGroupRfc.json --execution-
parameters file://CreateCDDepGroupParams.json
```

   Nella risposta riceverai l'ID RFC della nuova RFC. Salva l'ID per i passaggi successivi.

6. Invia la RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

   Se la RFC ha esito positivo, non riceverai alcun output.

7. Controlla lo stato della RFC:

```
aws amscm get-rfc --rfc-id RFC_ID
```

Carica l'applicazione WordPress

Hai automaticamente accesso a qualsiasi istanza di bucket S3 che crei. Puoi accedervi tramite i tuoi Bastions (vedi Accesso alle istanze) o tramite la console S3 e caricare il pacchetto. CodeDeploy Il pacchetto deve essere disponibile per continuare a distribuire lo stack. L'esempio utilizza il nome del bucket creato in precedenza.

```
aws s3 cp wordpress/wordpress.zip s3://ACCOUNT_ID-codedeploy-bundles/
```

Distribuisci l' WordPress applicazione con CodeDeploy

Distribuisci l' CodeDeploy applicazione.

Una volta che hai il pacchetto di CodeDeploy applicazioni e il gruppo di distribuzione, usa questa RFC per distribuire l'applicazione.

DATI RICHIESTI:

- `VPC-ID`: Il VPC che stai utilizzando, dovrebbe essere lo stesso del VPC usato in precedenza.

- `CodeDeployApplicationName`: Usa il nome dell' CodeDeploy applicazione che hai creato in precedenza.
- `CodeDeployDeploymentGroupName`: utilizza il nome del gruppo di CodeDeploy distribuzione creato in precedenza.
- `S3Location`(dove hai caricato il pacchetto di applicazioni):`S3Bucket`: BucketName Quello che hai creato in precedenza `S3BundleType` e`S3Key`: il tipo e il nome del pacchetto che hai inserito nel tuo negozio S3.
- `ChangeTypeIdeChangeTypeVersion`: L'ID del tipo di modifica per questa procedura dettagliata è`ct-2edc3sd1sqmrb`, per scoprire la versione più recente, esegui questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-2edc3sd1sqmrb
```

1. Esporta lo schema JSON dei parametri di esecuzione per la distribuzione dell' CodeDeploy applicazione CT in un file nella cartella corrente; l'esempio lo chiama Deploy Params.json. CDApp

```
aws amscm get-change-type-version --change-type-id "ct-2edc3sd1sqmrb" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > DeployCDAppParams.json
```

2. Modificate il file JSON come segue; potete eliminare e sostituire il contenuto. Per`S3Bucket`, usa `BucketName` quello che hai creato in precedenza.

```
{
"Description":                        "Deploy WordPress CodeDeploy Application",
"VpcId":                              "VPC_ID",
"Name":                               "WP CodeDeploy Deployment Group",
"TimeoutInMinutes":                   60,
"Parameters":   {
    "CodeDeployApplicationName":         "WordPressCDApp",
    "CodeDeployDeploymentGroupName":     "WordPressCDDepGroup",
    "CodeDeployIgnoreApplicationStopFailures": false,
    "CodeDeployRevision": {
      "RevisionType": "S3",
      "S3Location": {
        "S3Bucket": "ACCOUNT_ID.BUCKET_NAME",
        "S3BundleType": "zip",
        "S3Key": "wordpress.zip" }
        }
```

```
    }
}
```

3. Esporta il modello JSON in un file nella cartella corrente; l'esempio lo chiama Deploy CDApp RFC.json: CreateRfc

```
aws amscm create-rfc --generate-cli-skeleton > DeployCDAppRfc.json
```

4. Modifica e salva il file Deploy CDApp RFC.json; puoi eliminare e sostituire il contenuto.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2edc3sd1sqmrb",
"Title":                "CD-Deploy-For-WP-Stack-RFC",
"RequestedStartTime":   "2017-04-28T22:45:00Z",
"RequestedEndTime":     "2017-04-28T22:45:00Z"
}
```

5. Crea la RFC, specificando il file dei parametri di esecuzione e il file Deploy Rfc: CDApp

```
aws amscm create-rfc --cli-input-json file://DeployCDAppRfc.json  --execution-
parameters file://DeployCDAppParams.json
```

   Riceverai il RfcId nuovo RFC nella risposta. Salva l'ID per i passaggi successivi.

6. Invia la RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

   Se la RFC ha esito positivo, non riceverai alcun output.

Convalida la distribuzione dell'applicazione

Passa all'endpoint (ELB CName) del sistema di bilanciamento del carico creato in precedenza, con il percorso distribuito:/. WordPress WordPress Ad esempio:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Annulla la distribuzione dell'applicazione

Per interrompere la distribuzione, inviate il Delete Stack CT allo stack di database RDS, all'application load balancer, al gruppo Auto Scaling, al bucket S3 e all'applicazione e al gruppo Code Deploy, sei in tutto. RFCs Inoltre, puoi inviare una richiesta di servizio per l'eliminazione degli snapshot RDS (vengono eliminati automaticamente dopo dieci giorni, ma costano una piccola somma). Raccogli lo stack IDs per tutti e poi segui questi passaggi.

Questa procedura dettagliata fornisce un esempio di utilizzo della console AMS per eliminare uno stack S3; questa procedura si applica all'eliminazione di qualsiasi stack utilizzando la console AMS.

> **Note**
>
> Se si elimina un bucket S3, è necessario prima svuotarlo degli oggetti.

DATI RICHIESTI:

- `StackId`: Lo stack da usare. Puoi trovarlo consultando la pagina AMS Console Stacks, disponibile tramite un link nel menu di navigazione a sinistra. Utilizzando l'API/CLI AMS SKMS, esegui il riferimento per l'API AMS SKMS, consulta la scheda Report nella Console AWS Artifact. Operazione (nella CLI). `list-stack-summaries`
- L'ID del tipo di modifica per questa procedura dettagliata è «1.0"`ct-0q0bic0ywqk6c`, per scoprire la versione più recente, esegui questo comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-0q0bic0ywqk6c
```

CREAZIONE IN LINEA:

- Esegui il comando create RFC con i parametri di esecuzione forniti in linea (evita le virgolette quando fornisci i parametri di esecuzione in linea). E

```
aws amscm create-rfc --change-type-id "ct-0q0bic0ywqk6c" --change-type-version "1.0"
 --title "Delete My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

- Invia la RFC utilizzando l'ID RFC restituito nell'operazione di creazione RFC. Fino all'invio, la RFC rimane nello `Editing` stato e non viene applicata.

```
aws amscm submit-rfc --rfc-id RFC_ID
```

- Monitora lo stato RFC e visualizza l'output di esecuzione:

```
aws amscm get-rfc --rfc-id RFC_ID
```

CREAZIONE DEL MODELLO:

1. Esporta il modello RFC in un file nella cartella corrente; l'esempio lo chiama DeleteStackRfc .json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteStackRfc.json
   ```

2. Modifica e salva il file.json. DeleteStackRfc Poiché l'eliminazione di uno stack ha un solo parametro di esecuzione, i parametri di esecuzione possono essere contenuti nel DeleteStackRfc file.json stesso (non è necessario creare un file JSON separato con parametri di esecuzione).

   Le virgolette interne dell'estensione ExecutionParameters JSON devono essere eliminate con una barra rovesciata (\). Esempio senza ora di inizio e fine:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0q0bic0ywqk6c",
   "Title":                "Delete-My-Stack-RFC"
   "ExecutionParameters":  "{
           \"StackId\":\"STACK_ID\"}"
   }
   ```

3. Crea la RFC:

   ```
   aws amscm create-rfc --cli-input-json file://DeleteStackRfc.json
   ```

   Riceverai RfcId il nuovo RFC nella risposta. Ad esempio:

   ```
   {
   "RfcId": "daaa1867-ffc5-1473-192a-842f6b326102"
   }
   ```

Salva l'ID per i passaggi successivi.

4. Invia la RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Se la RFC ha esito positivo, non riceverai alcuna conferma dalla riga di comando.

5. Per monitorare lo stato della richiesta e visualizzare l'output di esecuzione:

```
aws amscm get-rfc --rfc-id RFC_ID --query "Rfc.
{Status:Status.Name,Exec:ExecutionOutput}" --output table
```

Manutenzione delle applicazioni

Una volta implementata l'infrastruttura, la sfida è aggiornarla in modo coerente in tutti gli ambienti AMS, dal controllo qualità allo staging alla produzione.

Questa sezione fornisce una panoramica del processo di inserimento del carico di lavoro AMS e alcuni esempi di diversi metodi che è possibile utilizzare per mantenere aggiornato il livello dell'infrastruttura cloud.

Strategie di manutenzione delle applicazioni

Il modo in cui distribuisci le applicazioni influisce sul modo in cui le gestisci. Questa sezione fornisce alcune strategie per la manutenzione delle applicazioni.

Gli aggiornamenti dell'ambiente possono comportare le seguenti modifiche:

- Aggiornamenti di sicurezza
- Nuove versioni delle tue applicazioni
- Modifiche alla configurazione dell'applicazione
- Aggiornamenti alle dipendenze

> **Note**
>
> Per qualsiasi implementazione di applicazioni, indipendentemente dal metodo, invia sempre una richiesta di servizio in anticipo per far sapere ad AMS che intendi implementare un'applicazione.

Esempi di installazione di applicazioni immutabili e mutabili

Mutabilità delle istanze di calcolo	Metodo di installazione dell'app	AMI
Mutable	Con CodeDeploy	Fornito da AMS
	Manualmente	

Mutabilità delle istanze di calcolo	Metodo di installazione dell'app	AMI
	Con uno chef o un burattino, Pull Based	
	Con Ansible o Salt, basato su Push	
Non modificabile	Con un AMI dorato	Personalizzato (basato su AMS fornito)

Distribuzione mutabile con un'AMI CodeDeploy abilitata

AWS CodeDeploy è un servizio che automatizza le distribuzioni di codice su qualsiasi istanza, comprese le istanze Amazon e EC2 le istanze eseguite localmente. Puoi usarlo CodeDeploy con AMS per creare e distribuire un'applicazione. CodeDeploy Tieni presente che AMS fornisce un profilo di istanza predefinito per CodeDeploy le applicazioni.

- Amazon Linux (versione 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

Prima di CodeDeploy utilizzarlo per la prima volta, è necessario completare una serie di passaggi di configurazione:

1. Installa o aggiorna la CLI AWS
2. Crea un ruolo di servizio per AWS CodeDeploy, utilizzi il Service Role ARN nella distribuzione

IDs per tutte le opzioni CT sono disponibili nel Change Type Reference.

> **Note**
>
> Attualmente, è necessario utilizzare lo storage Amazon S3 con questa soluzione.

I passaggi di base sono descritti qui e la procedura è dettagliata nella Guida per l'utente AMS.

1. Crea un bucket di storage Amazon S3. CT: ct-1a68ck03fn98r. Il bucket S3 deve avere il controllo delle versioni abilitato (per informazioni su come eseguire questa operazione, consulta Enabling Bucket Versioning).
2. Mettici sopra i tuoi artefatti raggruppati. CodeDeploy Puoi farlo con la console Amazon S3 senza richiedere l'accesso tramite AMS. Oppure utilizzando una variante di questo comando:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Trova un `customer-` AMI AMS; usa uno dei due modi:
   - Console AMS: la pagina dei dettagli del VPC per il VPC pertinente
   - API AMS Per il riferimento all'API AMS SKMS, consulta la scheda Report nella Console AWS Artifact. o CLI: `aws amsskms list-amis`
4. Crea un gruppo Autoscaling (ASG). CT: ct-2tylseo8rxfsc. Specificate l'AMI AMS, impostate il load balancer in modo che abbia porte aperte, specificate `customer-mc-ec2-instance-profile ASGIAMInstanceProfile` per.
5. Crea la tua CodeDeploy applicazione. CT: ct-0ah3gwb9seqk2. I parametri includono il nome di un'applicazione, ad esempio. `WordpressProd`
6. Crea il tuo gruppo CodeDeploy di distribuzione. CT: ct-2gd0u847qd9d2. I parametri includono il nome CodeDeploy dell'applicazione, il nome ASG, il nome del tipo di configurazione e l'ARN del ruolo di servizio.
7. Distribuisci l'applicazione. CodeDeploy CT: ct-2edc3sd1sqmrb. I parametri includono il nome CodeDeploy dell'applicazione, il nome del tipo di configurazione, il nome del gruppo di distribuzione, il tipo di revisione e la posizione del bucket S3 in cui si trovano gli artefatti. CodeDeploy

Distribuzione mutabile, istanze applicative configurate manualmente e aggiornate

Questa strategia di distribuzione delle applicazioni consiste in un aggiornamento semplice e manuale delle istanze delle applicazioni. Questi sono i passaggi fondamentali.

IDs per tutte le opzioni CT sono disponibili nel Change Type Reference.

> **Note**
>
> Attualmente, è necessario utilizzare lo storage Amazon S3 con questa soluzione.

I passaggi di base sono descritti qui; le varie procedure sono dettagliate nella Guida per l'utente AMS.

1. Crea un bucket di storage Amazon S3. CT: ct-1a68ck03fn98r. Il bucket S3 deve avere il controllo delle versioni abilitato (per informazioni su come eseguire questa operazione, consulta Enabling Bucket Versioning).
2. Inserisci gli artefatti dell'applicazione in bundle (tutto ciò di cui l'applicazione ha bisogno per avviarsi e funzionare). Puoi farlo con la console Amazon S3 senza richiedere l'accesso tramite AMS. Oppure utilizzando una variante di questo comando:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Trova un AMI AMS, li avrai CodeDeploy tutti. Per trovare un'AMI «cliente» usa uno dei seguenti metodi:

   - Console AMS: la pagina dei dettagli del VPC per il VPC pertinente
   - API AMS Per il riferimento all'API AMS SKMS, consulta la scheda Report nella Console AWS Artifact. o CLI: `aws amsskms list-amis`

4. Crea un' EC2 istanza con quell'AMI. CT: ct-14027q0sjyt1h. Specificate l'AMI AMS, impostate un tag `Key=backup, Value=true` e `customer-mc-ec2-instance-profile` specificate il `InstanceProfile` parametro. Annotate l'ID dell'istanza restituito.
5. Richiedi l'accesso amministrativo all'istanza. CT: ct-1dmlg9g1l91h6. Avrai bisogno del nome di dominio completo per il tuo account. Se non sei sicuro di quale sia il tuo FQDN, puoi trovarlo nei seguenti modi:

   - Utilizzando la Console di gestione AWS per i servizi di directory (nella scheda Directory Name di sicurezza e identità).
   - Esecuzione di uno di questi comandi (return directory classes; DC+DC+DC=FQDN): Windows: o Linux:. `whoami /fqdn` `hostname --fqdn`

6. Accedi all'istanza, consulta Accesso alle istanze tramite Bastions nella Guida per l'utente AMS.
7. Scarica i file applicativi in bundle dal bucket S3 all'istanza.

8. Richiedi un backup immediato con una richiesta di servizio ad AMS, dovrai conoscere l'ID dell'istanza.
9. Quando devi aggiornare l'applicazione, carica nuovi file nel tuo bucket S3 e segui i passaggi da 3 a 8.

Distribuzione mutabile con un'AMI configurata con uno strumento di distribuzione basato su pull

Questa strategia si basa sul `InstanceUserData` parametro del Managed Services Create EC2 CT. Per ulteriori informazioni sull'utilizzo di questo parametro, vedere Configurazione delle istanze con dati utente. Questo esempio presuppone uno strumento di distribuzione delle applicazioni basato su pull come Chef o Puppet.

L' CodeDeploy agente è supportato su tutti gli AMS. AMIs Ecco l'elenco di quelli supportati AMIs:

- Amazon Linux (versione 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

IDs per tutte le opzioni CT sono disponibili nel Change Types Reference.

> **Note**
>
> Attualmente, è necessario utilizzare lo storage Amazon S3 con questa soluzione.

I passaggi di base sono descritti qui e la procedura è dettagliata nella Guida per l'utente AMS.

1. Crea un bucket di storage Amazon S3. CT: ct-1a68ck03fn98r. Il bucket S3 deve avere il controllo delle versioni abilitato (per informazioni su come eseguire questa operazione, consulta Enabling Bucket Versioning).
2. Mettici sopra i tuoi artefatti raggruppati. CodeDeploy Puoi farlo con la console Amazon S3 senza richiedere l'accesso tramite AMS. Oppure utilizzando una variante di questo comando:

```
aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
```

3. Trova un `customer-` AMI AMS; usa uno dei due modi:
   - Console AMS: la pagina dei dettagli del VPC per il VPC pertinente
   - API AMS Per il riferimento all'API AMS SKMS, consulta la scheda Report nella Console AWS Artifact. o CLI: `aws amsskms list-amis`
4. Crea un'istanza. EC2 CT: ct-14027q0sjyt1h; impostate un tag `Key=backup, Value=true` e utilizzate il `InstanceUserData` parametro per specificare un bootstrap e altri script (Chef/Puppet agente di download, ecc.) e includete le chiavi di autorizzazione necessarie. È possibile trovare un esempio in tal senso nella Guida per l'utente AMS, sezione Change Mangement, esempi di creazione di una distribuzione HA a due livelli. In alternativa, richiedi l'accesso e accedi all'istanza e configurala con gli elementi di implementazione necessari. Ricorda che i comandi di distribuzione basati su pull passano dagli agenti sulle istanze al server principale aziendale e potrebbero richiedere l'autorizzazione per passare attraverso i bastioni. Potrebbe essere necessaria una richiesta di servizio ad AMS per richiedere l'accesso ai group/AD gruppi di sicurezza senza bastioni.
5. Ripetere il passaggio 4 per creare un'altra EC2 istanza e configurarla con il server master dello strumento di distribuzione.
6. Quando devi aggiornare l'applicazione, utilizza lo strumento di distribuzione per distribuire gli aggiornamenti alle tue istanze.

Distribuzione mutabile con un'AMI configurata tramite uno strumento di distribuzione basato su push

Questa strategia si basa sul `InstanceUserData` parametro del Managed Services Create EC2 CT. Per ulteriori informazioni sull'utilizzo di questo parametro, vedere Configurazione delle istanze con dati utente. Questo esempio presuppone uno strumento di distribuzione delle applicazioni basato su pull come Chef o Puppet.

IDs per tutte le opzioni CT sono disponibili nel Change Type Reference.

> **Note**
>
> Attualmente, è necessario utilizzare lo storage Amazon S3 con questa soluzione.

I passaggi di base sono descritti qui e la procedura è dettagliata nella Guida per l'utente AMS.

1. Crea un bucket di storage Amazon S3. CT: ct-1a68ck03fn98r. Il bucket S3 deve avere il controllo delle versioni abilitato (per informazioni su come eseguire questa operazione, consulta Enabling Bucket Versioning).
2. Mettici sopra i tuoi artefatti raggruppati. CodeDeploy Puoi farlo con la console Amazon S3 senza richiedere l'accesso tramite AMS. Oppure utilizzando una variante di questo comando:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Trova un AMI AMS, li avrai CodeDeploy tutti. Per trovare un'AMI «cliente» usa uno dei seguenti metodi:

   - Console AMS: la pagina dei dettagli del VPC per il VPC pertinente
   - API AMS Per il riferimento all'API AMS SKMS, consulta la scheda Report nella Console AWS Artifact. o CLI: `aws amsskms list-amis`

4. Crea un'istanza. EC2 CT: ct-14027q0sjyt1h; imposta un tag e usa il `InstanceUserData` parametro per eseguire un bootstrap e altri script tra cui chiavi di autorizzazione`Key=backup, Value=true`, SALT stack (avvia un minion, per maggiori informazioni consulta Bootstrapping Salt su Linux EC2 con Cloud-Init) o Ansible (installa una coppia di chiavi: per maggiori informazioni consulta Getting Started with Ansible and Dynamic Amazon Inventory Management). EC2 In alternativa, richiedi l'accesso e accedi all'istanza e configurala con gli artefatti di distribuzione necessari. Ricorda che i comandi basati su push provengono dalla sottorete aziendale alle istanze e potrebbe essere necessario configurare l'autorizzazione per farli passare attraverso i bastioni. Potrebbe essere necessaria una richiesta di servizio ad AMS per richiedere l'accesso ai gruppi di sicurezza group/AD senza bastioni.
5. Ripetere il passaggio 4 per creare un'altra EC2 istanza e configurarla con il server master dello strumento di distribuzione.
6. Quando devi aggiornare l'applicazione, utilizza lo strumento di distribuzione per distribuire gli aggiornamenti alle tue istanze.

Implementazione immutabile con un'AMI dorata

Questa strategia utilizza un'AMI «dorata» che hai configurato per comportarsi come desideri che facciano tutte le tue istanze dell'applicazione. Ad esempio, le istanze create con questa AMI dorata si aggiungono automaticamente al dominio e al DNS corretti, si configurano automaticamente, si riavviano e avviano tutti i sistemi necessari. Quando desideri aggiornare le istanze delle tue applicazioni, ricrea la Golden AMI e implementa istanze applicative completamente nuove con essa.

L' CodeDeploy agente è supportato su tutti gli AMS. AMIs Ecco l'elenco di quelli supportati AMIs:

- Amazon Linux (versione 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

IDs per tutte le opzioni CT sono disponibili nel Change Type Reference.

> **Note**
>
> Attualmente, è necessario utilizzare lo storage Amazon S3 con questa soluzione.

1. Crea un bucket di storage Amazon S3. CT: ct-1a68ck03fn98r. Il bucket S3 deve avere il controllo delle versioni abilitato (per informazioni su come eseguire questa operazione, consulta Enabling Bucket Versioning).
2. Inserisci gli artefatti dell'applicazione in bundle (tutto ciò di cui l'applicazione ha bisogno per avviarsi e funzionare). Puoi farlo con la console Amazon S3 senza richiedere l'accesso tramite AMS. Oppure utilizzando una variante di questo comando:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Trova un `customer-` AMI AMS; usa uno dei due modi:

   - Console AMS: la pagina dei dettagli del VPC per il VPC pertinente
   - API AMS Per il riferimento all'API AMS SKMS, consulta la scheda Report nella Console AWS Artifact. o CLI: `aws amsskms list-amis`

4. Crea un' EC2 istanza con quell'AMI. CT: ct-14027q0sjyt1h. Specificate l'AMI AMS, impostate un tag `Key=backup, Value=true` e specificate `customer-mc-ec2-instance-profile` per`InstanceProfile`. Annotate l'ID di istanza restituito.
5. Richiedi l'accesso amministrativo all'istanza. CT: ct-1dmlg9g1l91h6. Avrai bisogno del nome di dominio completo per il tuo account. Se non sei sicuro di quale sia il tuo FQDN, puoi trovarlo nei seguenti modi:

   - Utilizzando la Console di gestione AWS per i servizi di directory (nella scheda Directory Name di sicurezza e identità).

- Esecuzione di uno di questi comandi (return directory classes; DC+DC+DC=FQDN): Windows: o Linux:. `whoami /fqdn` `hostname --fqdn`

6. Accedi all'istanza, consulta Accesso alle istanze nella Guida per l'utente AMS.
7. Scarica sull'istanza i file dell'applicazione in bundle dal tuo bucket S3. Configura l'istanza in modo che distribuisca automaticamente l'applicazione completamente funzionante all'avvio.
8. Crea l'AMI dorata sull'istanza. CT: ct-3rqqu43krekby. Per i dettagli, consulta AMI | Create.
9. Configura un gruppo Auto Scaling per creare nuove istanze utilizzando quell'AMI. CT: ct-2tylseo8rxfsc. Quando devi aggiornare la tua applicazione, segui questa procedura e richiedi ad AMS di aggiornare l'ASG per utilizzare la nuova AMI dorata; usa un CT Gestione | Altro | Altro | Aggiornamento per questo.

Strategie di aggiornamento

Esistono diverse strategie che è possibile utilizzare per aggiornare le applicazioni o le istanze nell'ambiente gestito da AMS.

- Downtime pianificato: questa semplice strategia prevede la pianificazione dell'orario in cui l'applicazione deve essere offline e aggiornata manualmente. A tale scopo, inviate una richiesta Management | Other | Other | Update CT (ct-0xdawir96cy7k) per interrompere le istanze richieste. Effettua gli aggiornamenti necessari, quindi invia un'altra richiesta Management | Other | Other | Update CT (ct-0xdawir96cy7k) per avviare le istanze.

- Blu/Verde: questa strategia richiede un ambiente ridondante (due ambienti completamente funzionanti) e la messa offline di un ambiente utilizzando gli aggiornamenti del Domain Name System (DNS) o del web firewall (WAF) per reindirizzare il traffico. Aggiorna un ambiente e poi reindirizza nuovamente per aggiornare l'altro ambiente.

  Per ulteriori informazioni, consulta AWS CodeDeploy Introduces Blue/Green Deployments.

- Aggiornamento continuo con nuova AMI: qui hai una nuova AMI che personalizzi (vedi Creazione di AMI) e poi richiedi che AMS la distribuisca nel tuo gruppo Auto Scaling. A tale scopo, utilizzate un Management | Other | Other | Update CT (ct-0xdawir96cy7k).

AWS Managed Services Resource Scheduler

Usa AWS Managed Services (AMS) Resource Scheduler per pianificare l'avvio e l'arresto automatici di AutoScaling gruppi, EC2 istanze Amazon e istanze RDS nel tuo account. Questo aiuta a ridurre i costi di infrastruttura laddove le risorse non sono destinate a funzionare 24 ore su 24, 7 giorni su 7. La soluzione si basa su Instance Scheduler on AWS, ma contiene funzionalità e personalizzazioni aggiuntive specifiche per le esigenze di AMS.

> **Note**
>
> Per impostazione predefinita, AMS Resource Scheduler non interagisce con risorse che non fanno parte di uno stack. AWS CloudFormation La risorsa deve far parte di uno stack che inizia con «stack-», «sc-» o «SC-». Per pianificare le risorse che non fanno parte di uno CloudFormation stack, puoi aggiornare il parametro dello stack Resource Scheduler a. `ScheduleNonStackResources` `Yes`

AMS Resource Scheduler utilizza periodi e pianificazioni:

- I periodi definiscono gli orari di esecuzione di Resource Scheduler, ad esempio l'ora di inizio, l'ora di fine e i giorni del mese.
- Le pianificazioni contengono i periodi definiti, insieme a configurazioni aggiuntive, come la finestra di manutenzione SSM, il fuso orario, l'impostazione di ibernazione e così via; e specificano quando le risorse devono essere eseguite, in base alle regole del periodo configurate.

È possibile configurare questi periodi e pianificazioni utilizzando i tipi di modifica automatici di AMS Resource Scheduler (). CTs

Per tutti i dettagli sulle impostazioni disponibili per AMS Resource Scheduler, consulta la documentazione corrispondente di AWS Instance Scheduler in Solution components. Per una visione dell'architettura della soluzione, consulta la documentazione corrispondente di AWS Instance Scheduler in Architecture overview.html.

Installazione di AMS Resource Scheduler

Per implementare AMS Resource Scheduler, utilizza il tipo di modifica automatica (CT): Deployment | AMS Resource Scheduler | Solution | Deploy (ct-0ywnhc8e5k9z5) per generare una RFC

che poi distribuisca la soluzione nel tuo account. Una volta eseguita la RFC, nel tuo account viene automaticamente fornito uno stack contenente le risorse AMS Resource Scheduler con configurazione predefinita. CloudFormation Per ulteriori informazioni sui tipi di modifica del Resource Scheduler, consulta AMS Resource Scheduler.

> **Note**
>
> Per scoprire se AMS Resource Scheduler è già distribuito nel tuo account, controlla la console AWS Lambda per quell'account e cerca la funzione Scheduler. AMSResource

Dopo aver installato AMS Resource Scheduler nel tuo account, ti consigliamo di rivedere la configurazione predefinita e, se necessario, personalizzare configurazioni come tag key, fuso orario, servizi pianificati e così via, in base alle tue preferenze. Per i dettagli sulle personalizzazioni consigliate, consulta la sezione successiva. Personalizzazione di AMS Resource Scheduler

Per effettuare le configurazioni personalizzate o semplicemente confermare la configurazione del Resource Scheduler,

Personalizzazione di AMS Resource Scheduler

Ti consigliamo di personalizzare le seguenti proprietà di AMS Resource Scheduler utilizzando i tipi di modifica di AMS Resource Scheduler aggiornati, vedi AMS Resource Scheduler.

- Nome tag: il nome del tag che Resource Scheduler utilizzerà per associare le pianificazioni delle istanze alle risorse. Il valore predefinito è Schedule.
- Servizi pianificati: un elenco separato da virgole di servizi che Resource Scheduler può gestire. Il valore predefinito è «ec2, rds, autoscaling». I valori validi sono «ec2", «rds» e «autoscaling»
- Fuso orario predefinito: Specificare il fuso orario predefinito da utilizzare per Resource Scheduler. Il valore predefinito è UTC.
- Usa CMK: un elenco separato da virgole di Amazon KMS Customer Managed Key (CMK) ARNs a cui è possibile concedere le autorizzazioni a Resource Scheduler.
- Utilizzo LicenseManager: è possibile concedere le autorizzazioni a un elenco separato da virgole di AWS Licence Manager per quel Resource Scheduler. ARNs

> **Note**
>
> Di tanto in tanto, AMS può rilasciare funzionalità e correzioni per mantenere aggiornato AMS Resource Scheduler nell'account dell'utente. Quando ciò accade, tutte le personalizzazioni apportate all'AMS Resource Scheduler vengono mantenute.

Utilizzo di AMS Resource Scheduler

Per configurare AMS Resource Scheduler dopo l'implementazione della soluzione, utilizza il Resource Scheduler automatizzato CTs per creare, eliminare, aggiornare e descrivere (ottenere dettagli sui) periodi di AMS Resource Scheduler (gli orari in cui viene eseguito Resource Scheduler) e pianificazioni (i periodi configurati e altre opzioni). Per un esempio di utilizzo dei tipi di modifica di AMS Resource Scheduler, consulta AMS Resource Scheduler.

Per selezionare le risorse da gestire con AMS Resource Scheduler, dopo la distribuzione e la creazione della pianificazione, utilizzi AMS Tag Create CTs per etichettare i gruppi di Auto Scaling, gli stack Amazon RDS e le risorse EC2 Amazon con la chiave di tag che hai fornito durante la distribuzione e la pianificazione definita come valore del tag. Dopo aver taggato le risorse, viene pianificato l'avvio o l'arresto delle risorse in base alla pianificazione del Resource Scheduler definita.

L'utilizzo di AMS Resource Scheduler non comporta costi aggiuntivi. Tuttavia, la soluzione ne utilizza diverse Servizi AWS e queste risorse ti verranno addebitate man mano che vengono utilizzate. Per ulteriori dettagli, consulta Panoramica dell'architettura.

Per disattivare AMS Resource Scheduler:

- Per la disattivazione o la disattivazione temporanea: invia una RFC utilizzando il tipo di modifica automatizzato Management | AMS Resource Scheduler | State | Disable change type (ct-14v49adibs4db)
- Per la rimozione permanente: invia una richiesta RFC di gestione | Altro | Altro | Aggiornamento (revisione richiesta) (ct-0xdawir96cy7k) che richiede la rimozione dal sistema di automazione delle versioni di Resource Scheduler

Stima dei costi di AMS Resource Scheduler

Per tenere traccia dei risparmi sui costi, AMS Resource Scheduler presenta un componente che calcola ogni ora i risparmi sui costi stimati per le risorse Amazon EC2 e RDS gestite dallo

scheduler. Questi dati sui risparmi sui costi vengono quindi pubblicati come CloudWatch metrica (`AMS/ResourceScheduler`) per aiutarti a tenerne traccia. Lo strumento di stima dei costi stima i risparmi solo sulle ore di funzionamento delle istanze. Non tiene conto di altri costi, come i costi di trasferimento dei dati associati a una risorsa.

Lo strumento di stima del risparmio sui costi è abilitato con Resource Scheduler. Funziona ogni ora e recupera i dati sui costi e sull'utilizzo da. AWS Cost Explorer In base a questi dati, calcola il costo orario medio per ogni tipo di istanza e quindi proietta il costo per un'intera giornata se l'istanza era in esecuzione senza pianificazione. Il risparmio sui costi è la differenza tra il costo previsto e il costo effettivo riportato da Cost Explorer per un determinato giorno.

Ad esempio, se l'istanza A è configurata con Resource Scheduler per essere eseguita dalle 9:00 alle 17:00, ovvero otto ore in un determinato giorno. Cost Explorer riporta il costo pari a $1 e l'utilizzo a 8. Il costo orario medio è quindi di 0,125 USD. Se l'istanza non è stata pianificata con Resource Scheduler, l'istanza verrà eseguita 24 ore su 24 in quel giorno. In tal caso, il costo sarebbe stato 24x0,125 = 3 USD. Resource Scheduler ti ha aiutato a ottenere un risparmio sui costi di 2$.

Affinché lo strumento di stima del risparmio dei costi recuperi i costi e l'utilizzo solo per le risorse gestite da Resource Scheduler da Cost Explorer, la chiave tag utilizzata da Resource Scheduler per indirizzare le risorse deve essere attivata come tag di allocazione dei costi nella dashboard di fatturazione. Se l'account appartiene a un'organizzazione, la chiave del tag deve essere attivata nell'account di gestione dell'organizzazione. Per informazioni su questa operazione, vedere Attivazione dei tag di allocazione dei costi definiti dall'utente e dei tag di allocazione dei costi definiti dall'utente

Dopo l'attivazione della chiave del tag come tag di allocazione dei costi, la AWS fatturazione inizia a tenere traccia dei costi e dell'utilizzo delle risorse gestite da Resource Scheduler e, una volta che i dati sono disponibili, lo strumento di stima dei costi inizia a calcolare i risparmi sui costi e a pubblicare i dati nello spazio dei nomi delle metriche in. `AMS/ResourceScheduler` CloudWatch

Suggerimenti per la stima dei costi

Cost Savings Estimator non prende in considerazione sconti come istanze riservate, piani di risparmio e così via, nel calcolo. L'Estimator prende i costi di utilizzo da Cost Explorer e calcola il costo medio orario per le risorse. Per maggiori dettagli, consulta Understanding your AWS Cost Datasets: A Cheat Sheet

Affinché lo strumento di stima del risparmio sui costi recuperi i costi e l'utilizzo solo per le risorse gestite da Resource Scheduler da Cost Explorer, la chiave tag utilizzata da Resource Scheduler

per indirizzare le risorse deve essere attivata come tag di allocazione dei costi nella dashboard di fatturazione. Se l'account appartiene a un'organizzazione, la chiave del tag deve essere attivata nell'account di gestione dell'organizzazione. Per informazioni su questa operazione, consulta Tag di allocazione dei costi definiti dall'utente. Se il tag di allocazione dei costi non è attivato, lo stimatore non è in grado di calcolare i risparmi e pubblicare la metrica, anche se è abilitata.

Le migliori pratiche di AMS Resource Scheduler

Pianificazione delle istanze Amazon EC2

- Il comportamento di chiusura dell'istanza deve essere impostato su `stop` e non su. `terminate` È preimpostato `stop` per le istanze create con il tipo di modifica automatica AMS Amazon EC2 Create (ct-14027q0sjyt1h) e può essere impostato per le istanze EC2 Amazon create con l'ingestione, impostando la proprietà su. AWS CloudFormation `InstanceInitiatedShutdownBehavior` `stop` Se il comportamento di chiusura delle istanze è impostato su`terminate`, le istanze termineranno quando Resource Scheduler le arresta e lo scheduler non sarà in grado di riavviarle.
- Le EC2 istanze Amazon che fanno parte di un gruppo Auto Scaling non vengono elaborate singolarmente da AMS Resource Scheduler, anche se sono contrassegnate.
- Se il volume root dell'istanza di destinazione è crittografato con una chiave master del cliente (CMK) KMS, è necessario aggiungere un'`kms:CreateGrant`autorizzazione aggiuntiva al ruolo IAM di Resource Scheduler, affinché lo scheduler possa avviare tali istanze. Per impostazione predefinita, questa autorizzazione non viene aggiunta al ruolo per una maggiore sicurezza. Se richiedi questa autorizzazione, invia una RFC con il tipo di modifica Management | AMS Resource Scheduler | Solution | Update e specifica un elenco separato da virgole ARNs del KMS. CMKs

Pianificazione dei gruppi di Auto Scaling

- AMS Resource Scheduler avvia o arresta la scalabilità automatica dei gruppi di Auto Scaling, non delle singole istanze del gruppo. Cioè, lo scheduler ripristina la dimensione del gruppo Auto Scaling (inizio) o imposta la dimensione su 0 (arresto).
- AutoScaling Gruppo di tag con il tag specificato e non le istanze all'interno del gruppo.
- Durante l'arresto, AMS Resource Scheduler memorizza i valori di capacità minima, desiderata e massima del gruppo Auto Scaling e imposta la capacità minima e desiderata su 0. Durante l'avvio, lo scheduler ripristina la dimensione del gruppo Auto Scaling com'era durante l'arresto. Pertanto, le istanze del gruppo Auto Scaling devono utilizzare una configurazione di capacità appropriata in

modo che la chiusura e il riavvio delle istanze non influiscano su alcuna applicazione in esecuzione nel gruppo Auto Scaling.

- Se il gruppo Auto Scaling viene modificato (la capacità minima o massima) durante un periodo di esecuzione, lo scheduler memorizza la nuova dimensione del gruppo Auto Scaling e la utilizza per ripristinare il gruppo al termine di una pianificazione di interruzioni.

Pianificazione delle istanze Amazon RDS

- Lo scheduler può scattare un'istantanea prima di arrestare le istanze RDS (non si applica al cluster Aurora DB). Questa funzionalità è attivata per impostazione predefinita con il parametro del modello Create RDS Instance Snapshot impostato su true. AWS CloudFormation Lo snapshot viene conservato fino alla successiva interruzione dell'istanza Amazon RDS e alla creazione di una nuova istantanea.

  Scheduler può utilizzare istanze start/stop Amazon RDS che fanno parte di un cluster o di un database Amazon RDS Aurora o in una configurazione con più zone di disponibilità (Multi-AZ). Tuttavia, verifica la limitazione di Amazon RDS quando lo scheduler non sarà in grado di interrompere l'istanza Amazon RDS, in particolare le istanze Multi-AZ. Per pianificare l'avvio o l'arresto di Aurora Cluster, usa il parametro del modello Schedule Aurora Clusters (l'impostazione predefinita è true). Il cluster Aurora (non le singole istanze all'interno del cluster) deve essere etichettato con la chiave tag definita durante la configurazione iniziale e il nome della pianificazione come valore del tag per pianificare quel cluster.

  Ogni istanza Amazon RDS ha una finestra di manutenzione settimanale durante la quale vengono applicate eventuali modifiche al sistema. Durante la finestra di manutenzione, Amazon RDS avvierà automaticamente le istanze che sono state interrotte per più di sette giorni per applicare la manutenzione. Tieni presente che Amazon RDS non interromperà l'istanza una volta completato l'evento di manutenzione.

  Lo scheduler consente di specificare se aggiungere la finestra di manutenzione preferita di un'istanza Amazon RDS come periodo di esecuzione alla sua pianificazione. La soluzione avvierà l'istanza all'inizio della finestra di manutenzione e la interromperà al termine della finestra di manutenzione se nessun altro periodo di esecuzione specifica che l'istanza deve essere eseguita e se l'evento di manutenzione è completato.

  Se l'evento di manutenzione non viene completato entro la fine della finestra di manutenzione, l'istanza verrà eseguita fino all'intervallo di pianificazione successivo al completamento dell'evento di manutenzione.

> **Note**
>
> Lo Scheduler non verifica che una risorsa sia avviata o interrotta. Effettua la chiamata API e va avanti. Se la chiamata API fallisce, registra l'errore per consentirne l'analisi.

Considerazioni sulla sicurezza delle applicazioni

La sicurezza delle applicazioni include la valutazione delle autorizzazioni necessarie per l'esecuzione dell'applicazione, delle regole del firewall e dei ruoli IAM da abilitare per l'accesso all'applicazione.

Per comprendere meglio la AWS sicurezza generale, consulta Best Practices for Security, Identity and Compliance.

Accesso per la gestione della configurazione

AWS Managed Services (AMS) cerca di fornirti un'infrastruttura priva di problemi in modo da non doverti preoccupare di problemi di sicurezza, problemi di patch, problemi di backup, ecc. A tal fine, AMS consiglia ruoli IAM minimi che consentono solo a un gruppo specifico o a un server master, se si utilizza uno strumento di distribuzione delle applicazioni, l'accesso alle istanze che eseguono l'applicazione.

Regole del firewall di accesso alle applicazioni

Proprio come il sistema operativo (OS), tutti gli accessi alle applicazioni devono essere regolati utilizzando i gruppi di Active Directory (AD). Utilizzando Amazon Relational Database Service (Amazon RDS) come esempio, è necessario interrompere il mirror (replica) per aggiungere un nuovo utente. L'approccio migliore consiste nel creare un gruppo in AD e aggiungerlo al momento della creazione del database. Avere i gruppi in AMS AD significa che è possibile creare CTs per l'accesso alle applicazioni. Per informazioni sulla strategia di raggruppamento ufficiale per AD, consulta Using Group Nesting Strategy — AD Best Practices for Group Strategy.

Per ulteriori informazioni sugli alberi di dominio e sui parent/child domini, consulta Come funzionano i domini e le foreste.

Le seguenti regole illustrano una soluzione appropriata per un trust basato su foreste multidominio con utenti che si trovano in domini figli.

Istanze Windows

Queste sono le regole da configurare per i controller di dominio principale e figlio di Windows.

Controller di dominio principale, Windows

DA: controller di dominio principali A: sottoreti Windows stack e servizi condivisi

Porta di origine	Porta di destinazione	Protocollo
88	49152 - 65535	TCP
389	49152 - 65535	UDP

DA: sottoreti dello stack, compresi i servizi condivisi A: controller di dominio Windows Forest Root

Porta di origine	Porta di destinazione	Protocollo
49152 - 65535	88	TCP
49152 - 65535	389	UDP

Controller di dominio secondario, Windows

DA: Controller di dominio secondari A: Controller di dominio Windows AWS

Porta di origine	Porta di destinazione	Protocollo
49152 - 65535	53	TCP
49152 - 65535	88	TCP
49152 - 65535	389	UDP

DA: Controller di dominio secondari A: sottoreti Windows stack e servizi condivisi

Porta di origine	Porta di destinazione	Protocollo
88	49152 - 65535	TCP
135	49152 - 65535	TCP

Porta di origine	Porta di destinazione	Protocollo
389	49152 - 65535	TCP
389	49152 - 65535	UDP
445	49152 - 65535	TCP
49152 - 65535	49152 - 65535	TCP

DA: sottoreti dello stack, compresi i servizi condivisi A: controller di dominio secondari Windows

Porta di origine	Porta di destinazione	Protocollo
49152 - 65535	88	TCP
49152 - 65535	135	TCP
49152 - 65535	389	TCP
49152 - 65535	389	UDP
49152 - 65535	445	TCP
49152 - 65535	49152 - 65535	TCP

Istanze Linux

Queste sono le regole da configurare per i controller di dominio Linux padre e figlio.

Tutti i test sono stati eseguiti utilizzando Amazon Linux. Sebbene l'intervallo di porte dinamico per Windows sia compreso tra 49152 e 65535, molti kernel Linux utilizzano l'intervallo di porte da 32768 a 61000. Esegui il comando seguente per visualizzare l'intervallo di porte IP.

```
cat /proc/sys/net/ipv4/ip_local_port_range
```

Controller di dominio principale, Linux

DA: Controller di dominio principali A: stack Linux e sottoreti di servizi condivisi

Porta di origine	Porta di destinazione	Protocollo
389	32768 - 61000	UDP
88	32768 - 61000	TCP

DA: sottoreti Stack, compresi i servizi condivisi A: Linux Forest Root Domain Controller

Porta di origine	Porta di destinazione	Protocollo
32768 - 61000	88	TCP
32768 - 61000	389	UDP

Controller di dominio secondario, Linux

DA: Controller di dominio secondari A: Controller di dominio AWS Linux

Porta di origine	Porta di destinazione	Protocollo
49152 - 65535	53	TCP
49152 - 65535	88	TCP
389	49152 - 65535	UDP
49152 - 65535	389	UDP

DA: Controller di dominio secondari A: stack Linux e sottoreti di servizi condivisi

Porta di origine	Porta di destinazione	Protocollo
88	32768 - 61000	TCP
389	32768 - 61000	UDP

DA: sottoreti dello stack, compresi i servizi condivisi A: controller di dominio secondario Linux

Porta di origine	Porta di destinazione	Protocollo
32768 - 61000	88	TCP
32768 - 61000	389	UDP

Gestione del traffico in uscita AMS

Per impostazione predefinita, la route con un CIDR di destinazione di 0.0.0.0/0 per le sottoreti AMS private e per le applicazioni personalizzate ha come destinazione un gateway NAT (Network Address Translation). I servizi AMS TrendMicro e l'applicazione delle patch sono componenti che devono disporre dell'accesso in uscita a Internet in modo che AMS sia in grado di fornire il proprio servizio e che i sistemi operativi possano ottenere aggiornamenti. TrendMicro

AMS supporta la deviazione del traffico in uscita verso Internet tramite un dispositivo di uscita gestito dal cliente a condizione che:

- Funziona come un proxy implicito (ad esempio trasparente).

  e

- Consente le dipendenze HTTP e HTTPS di AMS (elencate in questa sezione) per consentire l'applicazione di patch e la manutenzione continui dell'infrastruttura gestita da AMS.

Alcuni esempi sono:

- Il gateway di transito (TGW) ha una route predefinita che punta al firewall locale gestito dal cliente tramite la connessione AWS Direct Connect nell'account Multi-Account Landing Zone Networking.
- Il TGW ha una route predefinita che punta a un endpoint AWS nel Multi-Account Landing Zone egress VPC che sfrutta AWS PrivateLink, puntando a un proxy gestito dal cliente in un altro account AWS.
- Il TGW ha un percorso predefinito che punta a un firewall gestito dal cliente in un altro account AWS, con una connessione site-to-site VPN come allegato alla Multi-Account Landing Zone TGW.

AMS ha identificato le dipendenze AMS HTTP e HTTPS corrispondenti e sviluppa e perfeziona queste dipendenze su base continuativa. Vedi egressMgmt.zip. Oltre al file JSON, lo ZIP contiene un file README.

> **Note**
>
> - Queste informazioni non sono complete: alcuni siti esterni obbligatori non sono elencati qui.
> - Non utilizzare questo elenco nell'ambito di una lista negata o di una strategia di blocco.
> - Questo elenco è inteso come punto di partenza per un set di regole di filtraggio delle uscite, con l'aspettativa che vengano utilizzati strumenti di reporting per determinare con precisione dove il traffico effettivo diverge dall'elenco.

Per richiedere informazioni sul filtraggio del traffico in uscita, inviate un'e-mail al vostro CSDM: ams-csdm@amazon.com.

Gruppi di sicurezza

In AWS VPCs, i gruppi di sicurezza AWS agiscono come firewall virtuali, controllando il traffico per uno o più stack (un'istanza o un insieme di istanze). Quando uno stack viene lanciato, viene associato a uno o più gruppi di sicurezza, che determinano a quale traffico è consentito raggiungerlo:

- Per gli stack nelle sottoreti pubbliche, i gruppi di sicurezza predefiniti accettano il traffico proveniente da HTTP (80) e HTTPS (443) da tutte le posizioni (Internet). Gli stack accettano anche traffico SSH e RDP interno dalla rete aziendale e dai bastioni AWS. Questi stack possono quindi uscire attraverso qualsiasi porta verso Internet. Possono inoltre accedere alle sottoreti private e ad altri stack della sottorete pubblica.
- Gli stack delle sottoreti private possono passare a qualsiasi altro stack della sottorete privata e le istanze all'interno di uno stack possono comunicare completamente tra loro tramite qualsiasi protocollo.

> **Important**
>
> Il gruppo di sicurezza predefinito per gli stack nelle sottoreti private consente a tutti gli stack della sottorete privata di comunicare con altri stack in quella sottorete privata. Se si desidera limitare le comunicazioni tra gli stack all'interno di una sottorete privata, è necessario creare

nuovi gruppi di sicurezza che descrivano la restrizione. Ad esempio, se desiderate limitare le comunicazioni a un server di database in modo che gli stack di quella sottorete privata possano comunicare solo da un server di applicazioni specifico tramite una porta specifica, richiedete un gruppo di sicurezza speciale. In questa sezione viene descritto come eseguire questa operazione.

Gruppi di sicurezza predefiniti

MALZ

La tabella seguente descrive le impostazioni predefinite del gruppo di sicurezza in entrata (SG) per gli stack. L'SG si chiama "SentinelDefaultSecurityGroupPrivateOnly-VPC-ID», dove è *ID* un ID VPC nel tuo account di landing zone multi-account AMS. Tutto il traffico in uscita è consentito verso "mc-initial-garden-SentinelDefaultSecurityGroupPrivateOnly" tramite questo gruppo di sicurezza (è consentito tutto il traffico locale all'interno delle sottoreti dello stack).

Tutto il traffico in uscita è consentito a 0.0.0.0/0 da un secondo gruppo di sicurezza ». SentinelDefaultSecurityGroupPrivateOnly

> **Tip**
>
> Se scegli un gruppo di sicurezza per un tipo di modifica AMS, come EC2 creare o OpenSearch creare un dominio, utilizzerai uno dei gruppi di sicurezza predefiniti descritti qui o un gruppo di sicurezza creato da te. Puoi trovare l'elenco dei gruppi di sicurezza, per VPC, nella EC2 console AWS o nella console VPC.

Esistono gruppi di sicurezza predefiniti aggiuntivi che vengono utilizzati per scopi AMS interni.

Gruppi di sicurezza AMS predefiniti (traffico in entrata)

Tipo	Protocollo	Intervallo porte	Origine
Tutto il traffico	Tutti	Tutti	SentinelDefaultSecurityGroupPrivateOnly (limita il traffico in uscita ai membri dello stesso gruppo di sicurezza)

Tipo	Protocollo	Intervallo porte	Origine
Tutto il traffico	Tutti	Tutti	SentinelDefaultSecurityGroupPrivateOnlyEgressAll (non limita il traffico in uscita)
HTTP, HTTPS, SSH, RDP	TCP	80/443 (Fonte 0.0.0.0/0) L'accesso SSH e RDP è consentito dai bastioni	SentinelDefaultSecurityGroupPublic (non limita il traffico in uscita)
Bastioni MALZ:			
SSH	TCP	22	SharedServices VPC CIDR e DMZ VPC CIDR, oltre a soluzioni on-premise fornite dal cliente CIDRs
SSH	TCP	22	
RDP	TCP	3389	
RDP	TCP	3389	
Bastioni SALZ:			
SSH	TCP	22	mc-initial-garden- SG LinuxBastion
SSH	TCP	22	mc-initial-garden- LinuxBastion DMSG
RDP	TCP	3389	mc-initial-garden- WindowsBastion SG
RDP	TCP	3389	mc-initial-garden- WindowsBastion DMSG

SALZ

La tabella seguente descrive le impostazioni predefinite del gruppo di sicurezza in entrata (SG) per gli stack. L'SG è denominato "mc-initial-garden- SentinelDefaultSecurityGroupPrivateOnly -*ID*" dove *ID* è un identificatore univoco. Tutto il traffico in uscita è consentito verso "mc-initial-garden-SentinelDefaultSecurityGroupPrivateOnly" tramite questo gruppo di sicurezza (è consentito tutto il traffico locale all'interno delle sottoreti dello stack).

Tutto il traffico in uscita è consentito a 0.0.0.0/0 da un secondo gruppo di sicurezza "- -». mc-initial-garden SentinelDefaultSecurityGroupPrivateOnlyEgressAll *ID*

> **Tip**
>
> Se scegli un gruppo di sicurezza per un tipo di modifica AMS, come EC2 creare o OpenSearch creare un dominio, utilizzerai uno dei gruppi di sicurezza predefiniti descritti qui o un gruppo di sicurezza creato da te. Puoi trovare l'elenco dei gruppi di sicurezza, per VPC, nella EC2 console AWS o nella console VPC.

Esistono gruppi di sicurezza predefiniti aggiuntivi che vengono utilizzati per scopi AMS interni.

Gruppi di sicurezza AMS predefiniti (traffico in entrata)

Tipo	Protocollo	Intervallo porte	Origine
Tutto il traffico	Tutti	Tutti	SentinelDefaultSecurityGroupPrivateOnly (limita il traffico in uscita ai membri dello stesso gruppo di sicurezza)
Tutto il traffico	Tutti	Tutti	SentinelDefaultSecurityGroupPrivateOnlyEgressAll (non limita il traffico in uscita)
HTTP, HTTPS, SSH, RDP	TCP	80/443 (Fonte 0.0.0.0/0) L'accesso SSH e RDP è consentito dai bastioni	SentinelDefaultSecurityGroupPublic (non limita il traffico in uscita)
Bastioni MALZ:			
SSH	TCP	22	SharedServices VPC CIDR e DMZ VPC CIDR, oltre a soluzioni on-premise fornite dal cliente CIDRs
SSH	TCP	22	
RDP	TCP	3389	

Tipo	Protocollo	Intervallo porte	Origine
RDP	TCP	3389	
Bastioni SALZ:			
SSH	TCP	22	mc-initial-garden- SG LinuxBastion
SSH	TCP	22	mc-initial-garden- LinuxBastion DMSG
RDP	TCP	3389	mc-initial-garden- WindowsBastion SG
RDP	TCP	3389	mc-initial-garden- WindowsBastion DMSG

Creare, modificare o eliminare gruppi di sicurezza

È possibile richiedere gruppi di sicurezza personalizzati. Nei casi in cui i gruppi di sicurezza predefiniti non soddisfano le esigenze delle applicazioni o dell'organizzazione, è possibile modificare o creare nuovi gruppi di sicurezza. Tale richiesta sarebbe considerata obbligatoria per l'approvazione e verrebbe esaminata dal team operativo AMS.

Per creare un gruppo di sicurezza al di fuori degli stack VPCs, invia una RFC utilizzando il tipo di `Deployment | Advanced stack components | Security group | Create (review required)` modifica (ct-1oxx2g2d7hc90).

Per le modifiche ai gruppi di sicurezza Active Directory (AD), utilizza i seguenti tipi di modifica:

- Per aggiungere un utente: invia una RFC utilizzando Management | Directory Service | Utenti e gruppi | Aggiungi utente al gruppo [ct-24pi85mjtza8k]
- Per rimuovere un utente: invia una RFC utilizzando Management | Directory Service | Utenti e gruppi | Rimuovi utente dal gruppo [ct-2019s9y3nfml4]

> **Note**
>
> Quando si utilizza «review required» CTs, AMS consiglia di utilizzare l'opzione ASAP Scheduling (scegliere ASAP nella console, lasciare vuote le date di inizio e fine nell'API/CLI) in quanto CTs richiedono che un operatore AMS esamini la RFC ed eventualmente

comunichi con l'utente prima che possa essere approvata ed eseguita. Se li pianifichi RFCs, assicurati di attendere almeno 24 ore. Se l'approvazione non avviene prima dell'orario di inizio programmato, la RFC viene rifiutata automaticamente.

Trova gruppi di sicurezza

Per trovare i gruppi di sicurezza collegati a uno stack o a un'istanza, usa la EC2 console. Dopo aver trovato lo stack o l'istanza, puoi vedere tutti i gruppi di sicurezza ad esso collegati.

Per informazioni su come trovare i gruppi di sicurezza nella riga di comando e filtrare l'output, consulta `describe-security-groups`.

Appendice: Questionario di onboarding delle applicazioni

Utilizzate questo questionario per descrivere gli elementi e la struttura della distribuzione in modo che AMS possa determinare quali componenti dell'infrastruttura sono necessari. I requisiti di onboarding per le applicazioni Line-of-Business (LoB) sono significativamente diversi dalle applicazioni di prodotto, quindi questo questionario è progettato per rispondere a entrambi.

Argomenti

Riepilogo della distribuzione

Una descrizione della distribuzione. Ad esempio:

- Questo account è destinato alla distribuzione di un'applicazione Line-of-Business (LoB) (anziché alla distribuzione di un'applicazione di prodotto).
- L'implementazione prevede un ARP (proxy inverso autenticato) con scalabilità automatica all'interno della sottorete dell'account. public/DMZ
- I server Web e applicativi verranno distribuiti all'interno della sottorete privata dell'account.
- Un'istanza Amazon RDS (Amazon Relational Database Service) verrà inoltre distribuita all'interno della sottorete privata dell'account.
- I server (ARP, web, applicazioni, database, load balancer e così via) sono separati in gruppi di sicurezza distinti.
- L'account richiede un design HA (alta disponibilità) distribuito tra le zone di disponibilità (AZs), ovvero Multi-AZ.

Componenti di implementazione dell'infrastruttura

Quali sono tutti i diversi componenti che dovranno essere configurati per supportare l'applicazione?

- Regione: cosa Regione AWS o quali regioni sono necessarie?
- Alta disponibilità (HA): quali zone di disponibilità verranno utilizzate?
- Virtual Private Cloud (VPC): cos'è il blocco CIDR per il VPC?
- Quali istanze di server sono necessarie?
  - Authenticated Reverse Proxy (ARP): sistema operativo, AMI, tipo di istanza, ID di sottorete, gruppo di sicurezza, porta di ingresso?
  - Server Application Deployment Tool: sistema operativo, AMI, tipo di istanza, ID di sottorete, gruppo di sicurezza, porta di ingresso (Chef, Puppet) o porta di uscita (Ansible, Saltstack)?
  - Amazon RDS con MySQL: versione DB, tipo di utilizzo, classe di istanza, ID di sottorete, gruppo di sicurezza, ID istanza DB, dimensione dello storage, Multi-AZ, tipo di autenticazione, crittografia?
  - Archiviazione: la tua app è stateless? Hai bisogno di bucket S3? Hai bisogno di uno storage persistente? Hai bisogno della crittografia dei dati inattivi sui tuoi volumi EBS? È necessaria la crittografia DB?
  - Endpoint server esterni (al Managed Services VPC): SMTP? LDAP?
  - Requisiti di rete: filtraggio di rete (basato su gruppi di sicurezza?)? Ispezione del traffico Web (in entrata? in uscita?)?
- Etichettatura: quali tag devono essere utilizzati per raggruppare le risorse in raccolte logiche? Ad esempio, tutte le risorse per uno stack di applicazioni. Seleziona i tag in base al tuo caso d'uso, ad esempio per `backup=true` abilitare i backup. Inoltre, è necessario utilizzare il tag `name=value` affinché tutte EC2 le istanze create visualizzino un nome nella console.
- Gruppi di sicurezza:
  - Quali gruppi di sicurezza sono necessari?
  - Regole di accesso ai gruppi di sicurezza?
  - Regole di uscita dei gruppi di sicurezza?

Piattaforma di hosting delle applicazioni

Per la tua piattaforma di hosting delle applicazioni, considera i seguenti requisiti possibili:

- Database crittografati?
- Chiavi di crittografia gestite da chi?
- Tutti i dati in transito e in archivio sono crittografati?

- Tutti gli utenti accedono al sistema tramite HTTPS?
- Tutte le system-to-system interazioni sono state approvate dal tuo team addetto alle operazioni di sicurezza?

Modello di distribuzione delle applicazioni

Considerazioni su come si pianifica la distribuzione delle applicazioni. Per informazioni, consultare Qual è il mio modello operativo?.

- Automatizzata o manuale? Nessuna automazione dell'implementazione significa nessuna scalabilità automatica. Se richiedi l'accesso, accedi e aggiorni manualmente l'applicazione, l'aggiornamento non riesce. AMS si aspetta che tu ripristini l'aggiornamento o ci avvisi tramite una richiesta di assistenza in modo da poterti aiutare.
- Se automatizzato, qual è il framework? Script? Basato su agenti ()? puppet/chef)? Agentless (SALT/Ansible CodeDeploy? Gli strumenti di distribuzione basati su agenti e senza agenti richiedono la creazione e la distribuzione di un'istanza separata come server principale per gli strumenti. AMS si aspetta che tu conosca tutti gli elementi necessari per una corretta implementazione degli strumenti di implementazione delle applicazioni; tuttavia, siamo lieti di aiutarti con le relative domande sull'infrastruttura.
- Le vostre Line-of-Business applicazioni (quelle che utilizzate per creare e gestire le vostre applicazioni) richiedono l'applicazione di patch?

Dipendenze delle applicazioni

Avete bisogno di istanze per applicazioni Line-of-Business (LoB)? Per applicazioni di prodotto?

Di cosa hanno bisogno le applicazioni del prodotto per funzionare correttamente?

- Dipendenze a livello di rete: ad esempio, AWS Direct Connect
- Dipendenze dei pacchetti: ad esempio, `pip`
- Applicazioni da cui dipende questa applicazione: ad esempio, MySql
- Dipendenze dal firewall?

Di cosa hanno bisogno le vostre applicazioni LoB per funzionare correttamente?

- Dipendenze a livello di rete: ad esempio, AWS Direct Connect

- Dipendenze dei pacchetti: ad esempio, Firefox Saucy
- Applicazioni da cui dipende questa applicazione: ad esempio MySql
- Dipendenze dal firewall?

Certificati SSL per applicazioni di prodotto

Di quali certificati SSL avranno bisogno i vostri server in modo che le vostre applicazioni (LoB e prodotto) possano raggiungere tutto ciò di cui hanno bisogno per funzionare ed essere accessibili?

- Gruppo Auto Scaling?
- Database (Amazon RDS)?
- Load Balancer?
- Server degli strumenti di distribuzione?
- Firewall per applicazioni Web (AWS WAF)?
- Altre istanze?

Ad esempio, per ciascuna delle istanze sopra elencate potrebbero essere necessari i seguenti certificati:

WAF (certificato 1) -> ELB-ext (certificato 2) -> ARP (certificato 3) -> ELB-int (certificato 4) -> Sito Web (certificato 5) -> Elb-int (certificato 6) -> Servizio Web (certificato 7).

Cronologia dei documenti

La tabella seguente descrive la documentazione per questa versione di AMS.

- Versione API: 2019-05-21
- Ultimo aggiornamento della documentazione: 16 febbraio 2023

Modifica	Descrizione	Link
Link TOC rimosso	Link al AWS glossario TOC rimosso.	08 agosto 2025
Contenuto aggiornato: Migrazione dei carichi di lavoro: convalida pre-ingestione di Windows	Sezione aggiornata che include passaggi dettagliati per l'utilizzo dello script di WIGs pre-validazione per verificare che l'istanza di Windows sia pronta per l'inserimento nell'account AMS;.	Migrazione dei carichi di lavoro: convalida pre-ingestione di Windows
Contenuto aggiornato, configurazione DMS	una nota importante sul ruolo richiesto, dms-vpc-role.	1: sottogruppo AWS DMS di replica: creazione
Contenuti aggiornati, risorse supportate da CFN Ingest	Aggiunto. OpenSearch	Risorse supportate
Contenuti aggiornati, migrazione dei carichi di lavoro	Istruzioni aggiornate per la convalida prima dell'ingestione.	Migrazione dei carichi di lavoro: convalida pre-ingestione di Windows

Modifica	Descrizione	Link
Contenuti aggiornati, CFN Ingest.	Sono state rimosse le «risorse supportate» con restrizioni dai contenuti di CFN ingest.	CloudFormation Ingest Stack: risorse supportate
Versioni Windows supportate aggiornate	È stato aggiunto il supporto per Windows Server 2022.	Immagini di macchine AMS Amazon (AMIs), Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows e Migrazione dei carichi di lavoro: convalida pre-ingestione di Windows
Contenuto aggiornato, Resource Scheduler.	Istruzioni aggiornate per l'uso del CT di implementazione dedicato, ct-0ywnhc8e5k9z5, applicabile sia a SALZ che a MALZ.	Avvio rapido di AMS Resource Scheduler
Contenuti aggiornati, Workload Ingest.	Versioni SUSE Linux supportate aggiornate.	Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows

Modifica	Descrizione	Link
Contenuti aggiornati, Database Migration Service.	Aggiunto ai prerequisiti e apportato diverse modifiche per quanto riguarda l'utilità e l'usabilità.	AWS Database Migration Service (AWS DMS)
Contenuti aggiornati, Workload Ingest.	Il Linux Pre-WIGS Validation Zip è stato aggiornato.	Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows
Contenuti aggiornati.	Aggiornato lo zip di convalida precedente a WIGS per Linux. Inoltre, è stato aggiunto Windows Server 2008 R2 come sistema operativo supportato.	Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows
Nuovo contenuto	I Quick Start e i tutorial sono stati spostati qui dalla versione non più disponibile della AMS Advanced Change Management Guide.	Avviamenti rapidi, Tutorial.
Contenuti aggiornati	Distribuzione | Componenti stack avanzati | Database Migration Service (DMS) | Avvia attività di replica (ct-1yq7hhqse71yg) Aggiornato per indicare che i parametri e Region sono obbligatori; in precedenza, erano erroneamente elencati come DocumentNameopzionali.	Database Migration Service (DMS) | Avvia attività di replica

Modifica	Descrizione	Link
Contenuti aggiornati	CloudFormation Ingerisci Aggiornato per indicare due nuove risorse supportate AWS::Route53Resolver::ResolverRuleAssociation e AWS::Route53Resolver::ResolverRule.	Risorse supportate
Contenuti aggiornati	Migrazione dei carichi di lavoro: convalida pre-ingestione di Windows	Informazioni su Sysprep aggiornate con ulteriori dettagli. Migrazione dei carichi di lavoro: convalida pre-ingestione di Windows
Contenuti aggiornati	Gestione | Stack personalizzato | Stack from CloudFormation Template | Approva Changeset e aggiorna (ct-1404e21baa2ox) La descrizione dettagliata CT del parametro è stata aggiornata con informazioni aggiuntive. ChangeSetName	Stack from CloudFormation Template | Approva il changeset e aggiorna
	Disponibili Ubuntu 18.04 e Oracle Linux 8.3	Migrazione dei carichi di lavoro: prerequisiti per Linux e Windows

Modifica	Descrizione	Link
Nuovi contenuti:	Implementazioni IAM tramite CFN Ingest e Stack Update. CTs	10 febbraio 2022
Attività di replica del Database Migration Service (DMS)	I tipi di modifica sono stati aggiornati in modo che le espressioni regolari ARNs consentano attività che contengono trattini. Avviare l' AWS DMS attività di replicae Database Migration Service (DMS) | Stop Replication Task.	13 gennaio 2022
Convalida pre-ingestion di Linux WIGS	Il file zip è stato aggiornato. Migrazione dei carichi di lavoro: convalida pre-ingestione di Linux.	13 gennaio 2022
Collegamenti fissi	La Configurazione sezione Importazione del database (DB) in AMS SQL RDS -> presentava alcuni collegamenti errati.	13 gennaio 2022

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.