aws

Best Practice Guida

Amazon Elastic Container Service

Amazon Elastic Container Service: Best Practice Guida

Table of Contents

Introduction

Amazon Elastic Container Service (Amazon ECS) è un servizio rapido e altamente scalabile di gestione dei container che consente di eseguire, arrestare e gestire container in un cluster. Questa guida illustra molte delle best practice operative più importanti, spiegando al contempo gli argomenti fondamentali su come funzionano le applicazioni basate su Amazon ECS. L'obiettivo è fornire un approccio concreto e fruibile al funzionamento e alla risoluzione dei problemi delle applicazioni basate su Amazon ECS.

Questa guida verrà rivista regolarmente per incorporare le nuove best practice Amazon ECS. Se hai domande o commenti su uno qualsiasi dei contenuti di questa guida, sollevi un problema nel repository GitHub. Per ulteriori informazioni, consultaGuida alle best practice di Amazon ECSsu GitHub

- Best practice - Networking
- Best Practice - Scalabilità automatica e gestione della capacità
- Procedure ottimali - Archiviazione persistente
- Best practice - Sicurezza

Best practice - Networking

Le applicazioni moderne sono in genere costituite da più componenti distribuiti che comunicano tra loro. Ad esempio, un'applicazione mobile o Web potrebbe comunicare con un endpoint API e l'API potrebbe essere alimentata da più microservizi che comunicano su Internet.

Questa guida illustra le procedure consigliate per la creazione di una rete in cui i componenti dell'applicazione possano comunicare tra loro in modo sicuro e scalabile.

Argomenti

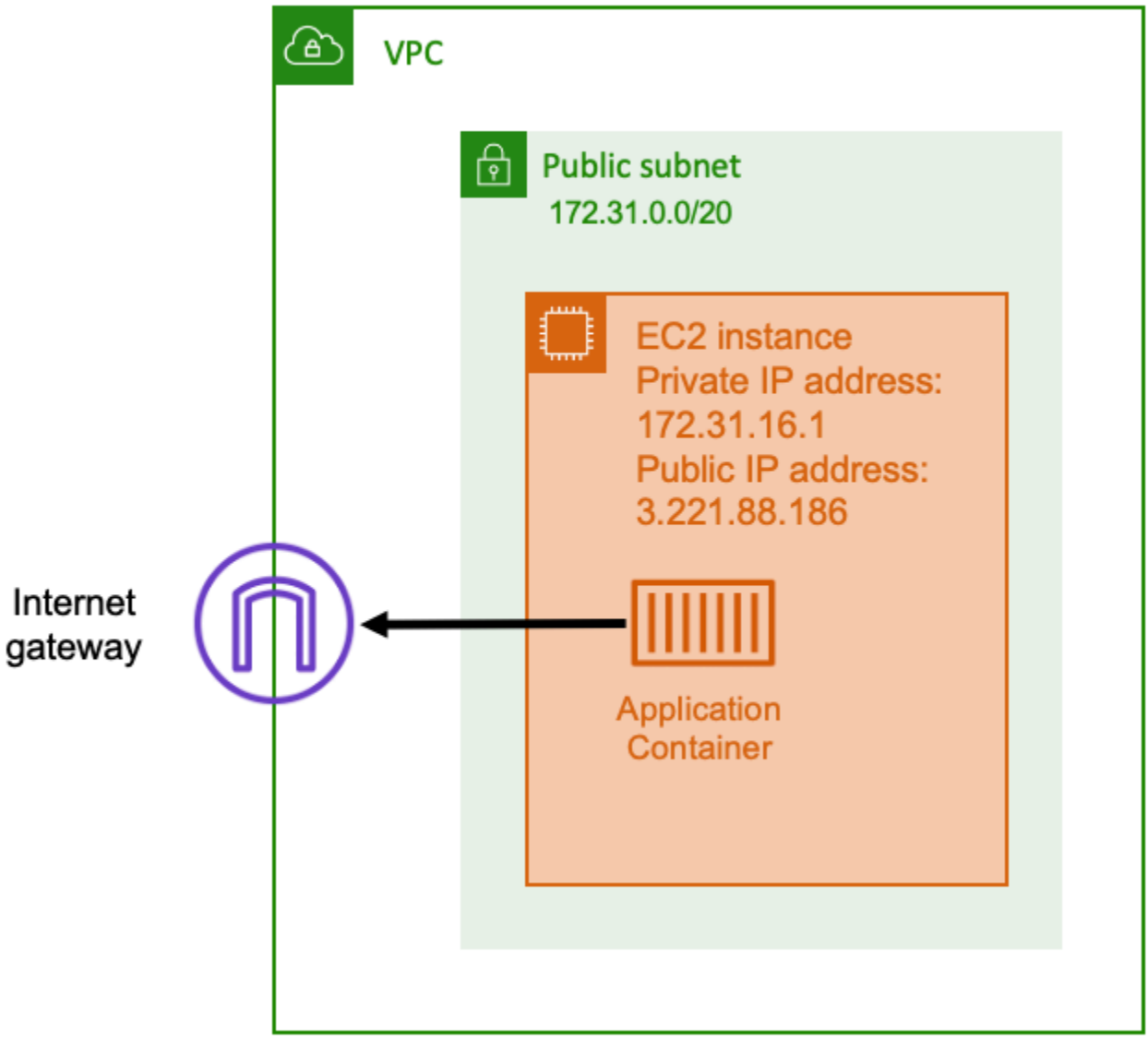
Connessione a Internet

La maggior parte delle applicazioni containerizzate dispone di almeno alcuni componenti che necessitano di accesso in uscita a Internet. Ad esempio, il back-end per un'app mobile richiede l'accesso in uscita alle notifiche push.

Amazon Virtual Private Cloud ha due metodi principali per facilitare la comunicazione tra il VPC e Internet.

Argomenti

Utilizzo di una subnet pubblica e di un gateway Internet



Utilizzando una sottorete pubblica che ha una route a un Internet Gateway, l'applicazione containerizzata può essere eseguita su un host all'interno di un VPC in una sottorete pubblica. All'host che esegue il contenitore viene assegnato un indirizzo IP pubblico. Questo indirizzo IP pubblico è instradabile da Internet. Per ulteriori informazioni, consultaGateway InternetnellaGuida per l'utente di Amazon VPC: .

Questa architettura di rete facilita la comunicazione diretta tra l'host che esegue l'applicazione e altri host su Internet. La comunicazione è bidirezionale. Ciò significa che non solo è possibile stabilire una connessione in uscita a qualsiasi altro host su Internet, ma anche altri host su Internet potrebbero

tentare di connettersi al proprio host. Pertanto, è necessario prestare particolare attenzione alle regole del gruppo di sicurezza e del firewall. Questo per garantire che gli altri host su Internet non possano aprire connessioni che non si desidera aprire.

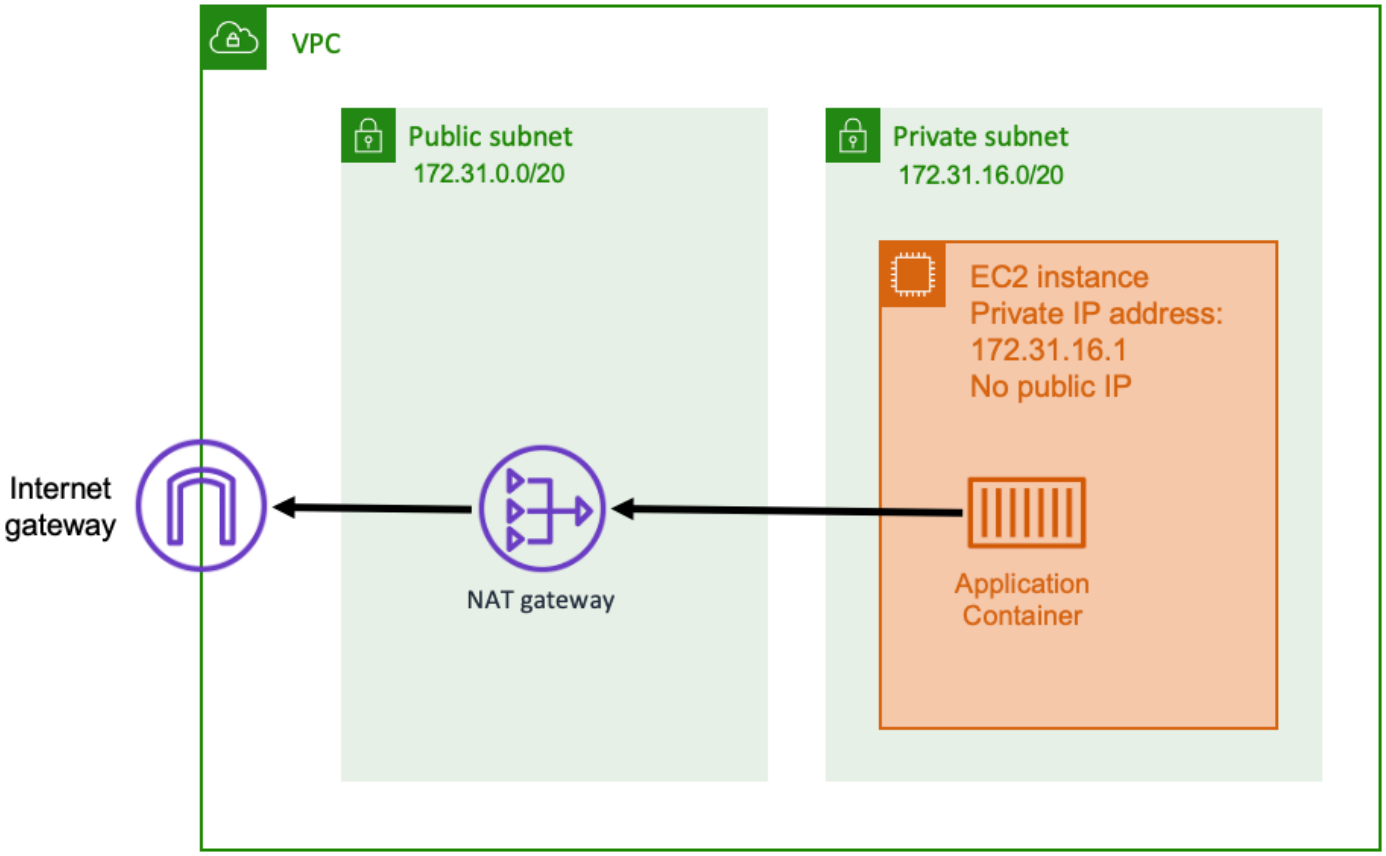
Ad esempio, se l'applicazione è in esecuzione su Amazon EC2, assicurati che la porta 22 per l'accesso SSH non sia aperta. In caso contrario, la tua istanza potrebbe ricevere tentativi di connessione SSH costanti da bot macilious su Internet. Questi bot traina attraverso indirizzi IP pubblici. Dopo aver trovato una porta SSH aperta, tentano di forzare le password brute-force per tentare di accedere all'istanza. Per questo motivo, molte organizzazioni limitano l'utilizzo delle subnet pubbliche e preferiscono avere la maggior parte, se non tutte, delle loro risorse all'interno di subnet private.

L'utilizzo di subnet pubbliche per la rete è adatto per applicazioni pubbliche che richiedono grandi quantità di larghezza di banda o latenza minima. I casi d'uso applicabili includono lo streaming video e i servizi di gioco.

Questo approccio di rete è supportato sia quando utilizzi Amazon ECS su Amazon EC2 che quando lo usi suAWS Fargate: .

- Utilizzando Amazon EC2: puoi avviare istanze EC2 su una sottorete pubblica. Amazon ECS utilizza queste istanze EC2 come capacità del cluster e tutti i contenitori in esecuzione sulle istanze possono utilizzare l'indirizzo IP pubblico sottostante dell'host per la rete in uscita. Questo vale sia per il`hostebridge`modalità di rete. Tuttavia, il`awsvpc`La modalità di rete non fornisce alle ENI delle attività gli indirizzi IP pubblici. Pertanto, non possono utilizzare direttamente un gateway Internet.
- Utilizzo di Fargate: quando crei il tuo servizio Amazon ECS, specifica le subnet pubbliche per la configurazione di rete del tuo servizio e assicurati che ilAssegna indirizzo IP pubblicoÈ abilitata. Ogni attività di Fargate è collegata in rete nella subnet pubblica e dispone di un proprio indirizzo IP pubblico per la comunicazione diretta con Internet.

Utilizzo di una subnet privata e di un gateway NAT



Utilizzando una subnet privata e un gateway NAT, è possibile eseguire l'applicazione containerizzata su un host che si trova in una subnet privata. Pertanto, questo host ha un indirizzo IP privato che è instradabile all'interno del VPC, ma non è instradabile da Internet. Ciò significa che altri host all'interno del VPC possono effettuare connessioni all'host utilizzando il suo indirizzo IP privato, ma altri host su Internet non possono effettuare comunicazioni in ingresso con l'host.

Con una sottorete privata, puoi utilizzare un gateway NAT (Network Address Translation) per consentire a un host all'interno di una sottorete privata di connettersi a Internet. Gli host su Internet ricevono una connessione in ingresso che sembra provenire dall'indirizzo IP pubblico del gateway NAT che si trova all'interno di una subnet pubblica. Il gateway NAT è responsabile di fungere da ponte tra Internet e il VPC privato. Questa configurazione è spesso preferita per motivi di sicurezza perché significa che il VPC è protetto dall'accesso diretto da parte di utenti malintenzionati su Internet. Per ulteriori informazioni, consultaGateway NATnellaGuida per l'utente di Amazon VPC: .

Questo approccio di rete privata è adatto per scenari in cui si desidera proteggere i contenitori dall'accesso esterno diretto. Gli scenari applicabili includono sistemi di elaborazione dei pagamenti

o contenitori che memorizzano i dati utente e le password. Ti vengono addebitati solo i costi di creazione e di utilizzo di un gateway NAT nel tuo account. Si applicano inoltre le tariffe orarie di utilizzo ed elaborazione dati del gateway NAT. Per motivi di ridondanza, è necessario disporre di un gateway NAT in ogni zona di disponibilità. In questo modo, la perdita di disponibilità di una singola zona di disponibilità non compromette la connettività in uscita. Per questo motivo, se si dispone di un carico di lavoro ridotto, potrebbe risultare più conveniente utilizzare subnet private e gateway NAT.

Questo approccio di rete è supportato sia quando si utilizza Amazon ECS su Amazon EC2 che quando lo si utilizza suAWS Fargate: .

- Utilizzando Amazon EC2: puoi avviare istanze EC2 su una sottorete privata. I contenitori eseguiti su questi host EC2 utilizzano la rete host sottostanti e le richieste in uscita passano attraverso il gateway NAT.
- Utilizzo di Fargate: quando crei il tuo servizio Amazon ECS, specifica le subnet private per la configurazione di rete del tuo servizio e non attiva l'opzioneAssegna indirizzo IP pubblicoopzione. Ogni attività di Fargate è ospitata in una sottorete privata. Il traffico in uscita viene instradato attraverso qualsiasi gateway NAT associato a tale subnet privata.

Ricezione di connessioni in entrata da Internet

Se si esegue un servizio pubblico, è necessario accettare il traffico in entrata da Internet. Ad esempio, il sito Web pubblico deve accettare richieste HTTP in ingresso dai browser. In tal caso, anche altri host su Internet devono avviare una connessione in ingresso all'host dell'applicazione.

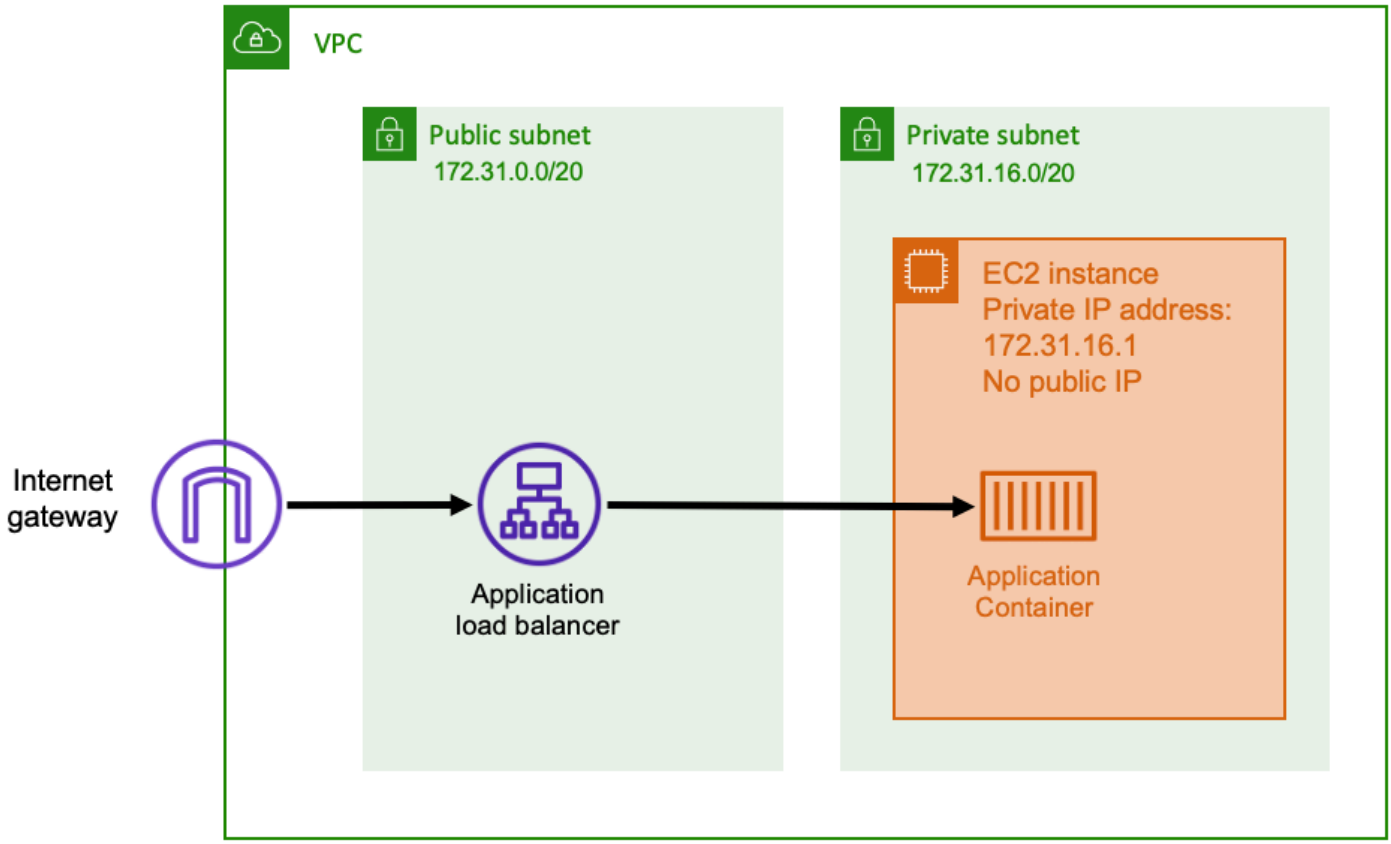
Un approccio a questo problema consiste nell'avviare i contenitori su host che si trovano in una subnet pubblica con un indirizzo IP pubblico. Tuttavia, non è consigliabile utilizzarlo per applicazioni su larga scala. Per questi, un approccio migliore è avere un livello di input scalabile che si trova tra Internet e l'applicazione. Per questo approccio puoi utilizzare uno qualsiasi deiAWSelencati in questa sezione come input.

Argomenti

- Application Load Balancer
- Network Load Balancer
- API HTTP Amazon API Gateway

Application Load Balancer

Un Application Load Balancer funziona a livello di applicazione. È il settimo livello del modello Open Systems Interconnection (OSI). Questo rende un Balancer Application Load Balancer adatto per i servizi HTTP pubblici. Se si dispone di un sito Web o di un'API REST HTTP, un servizio di Application Load Balancer è un servizio di bilanciamento del carico adatto per questo carico di lavoro. Per ulteriori informazioni, consultaCos'è un Application Load Balancer?nellaGuida per l'utente dei sistemi Application Load Balancer: .



Con questa architettura, è possibile creare un servizio di Application Load Balancer in una subnet pubblica in modo che disponga di un indirizzo IP pubblico e possa ricevere connessioni in ingresso da Internet. Quando Application Load Balancer riceve una connessione in ingresso o più specificamente una richiesta HTTP, apre una connessione all'applicazione utilizzando il suo indirizzo IP privato. Quindi, inoltra la richiesta tramite la connessione interna.

Un Application Load Balancer presenta i vantaggi seguenti.

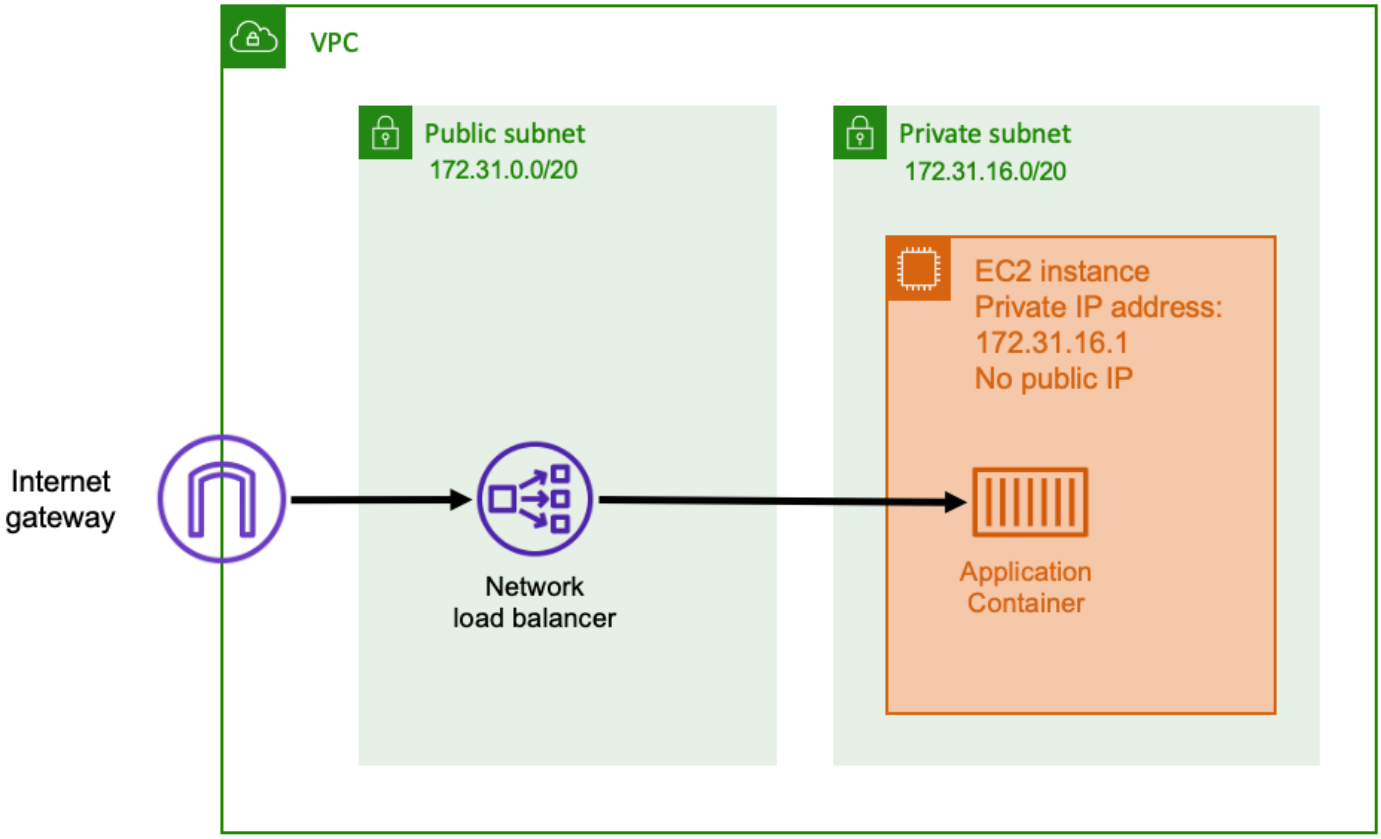
- Terminazione SSL/TLS: un Application Load Balancer può sostenere comunicazioni HTTPS sicure e certificati per le comunicazioni con i client. Può opzionalmente terminare la connessione

SSL a livello di bilanciamento del carico in modo da non dover gestire i certificati nella propria applicazione.

- Routing avanzato: un Application Load Balancer può avere più nomi host DNS. Dispone inoltre di funzionalità di routing avanzate per inviare richieste HTTP in ingresso a destinazioni diverse in base a metriche come il nome host o il percorso della richiesta. Ciò significa che è possibile utilizzare un singolo Application Load Balancer come input per molti servizi interni diversi o anche microservizi su percorsi diversi di un'API REST.
- Supporto gRPC e websocket: un Application Load Balancer è in grado di gestire più di un semplice HTTP. Può anche bilanciare il carico gRPC e servizi basati su websocket, con supporto HTTP/2.
- Sicurezza: un servizio di Application Load Balancer aiuta a proteggere l'applicazione dal traffico dannoso. Include funzionalità come le attenuazioni di sincronizzazione HTTP ed è integrato con AWS Web Application Firewall (AWS WAF).AWS WAFpuò filtrare ulteriormente il traffico dannoso che potrebbe contenere modelli di attacco, ad esempio SQL injection o cross-site scripting.

Network Load Balancer

Un Network Load Balancer funziona al quarto livello del modello Open Systems Interconnection (OSI). È adatto per protocolli non HTTP o scenari in cui è necessaria la crittografia end-to-end, ma non ha le stesse caratteristiche specifiche di HTTP di un Application Load Balancer. Di conseguenza, un servizio di Network Load Balancer è più adatto per le applicazioni che non utilizzano HTTP. Per ulteriori informazioni, consultaCos'è un servizio di Network Load Balancer?nellaGuida per l'utente dei sistemi Network Load Balancer: .



Quando un servizio di Network Load Balancer viene utilizzato come input, funziona in modo simile a un Application Load Balancer. Questo perché è stato creato in una subnet pubblica e dispone di un indirizzo IP pubblico a cui è possibile accedere su Internet. Il servizio di Network Load Balancer apre quindi una connessione all'indirizzo IP privato dell'host che esegue il contenitore e invia i pacchetti dal lato pubblico al lato privato.

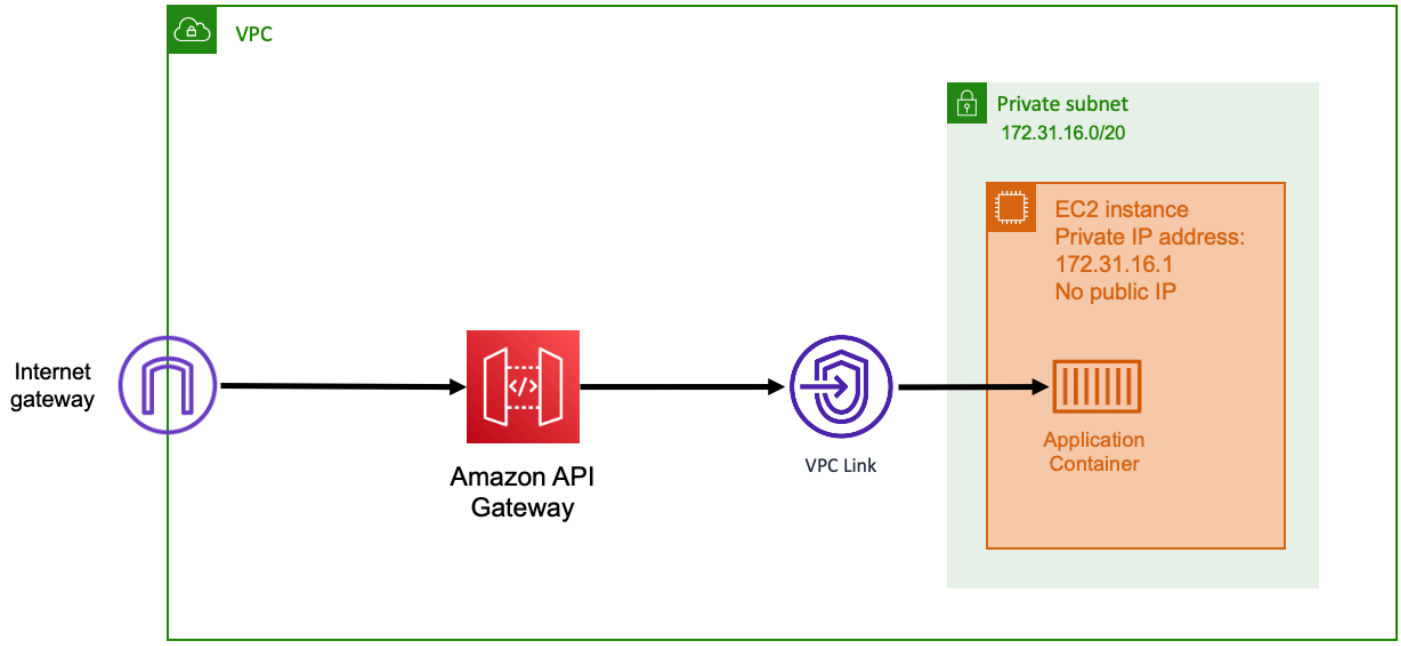
Poiché Network Load Balancer opera a un livello inferiore dello stack di rete, non dispone dello stesso set di funzionalità di Application Load Balancer. Tuttavia, ha le seguenti caratteristiche importanti.

- Crittografia end-to-end — Poiché un Network Load Balancer opera al quarto livello del modello OSI, non legge il contenuto dei pacchetti. Ciò lo rende adatto per comunicazioni di bilanciamento del carico che richiedono crittografia end-to-end.
- Crittografia TLS: oltre alla crittografia end-to-end, Network Load Balancer può anche terminare le connessioni TLS. In questo modo, le applicazioni back-end non devono implementare il proprio TLS.

- Supporto UDP — Poiché un Network Load Balancer opera al quarto livello del modello OSI, è adatto per carichi di lavoro e protocolli non HTTP diversi da TCP.

API HTTP Amazon API Gateway

Amazon API Gateway HTTP API è un server meno in ingresso adatto per applicazioni HTTP con improvvisi raffiche nei volumi di richiesta o volumi di richiesta bassi. Per ulteriori informazioni, consultaChe cosa è Amazon API Gateway?nellaGuida per sviluppatori di API Gateway: .



Il modello di determinazione dei prezzi per Application Load Balancer e Network Load Balancer include un prezzo orario per mantenere i bilanciatori disponibili per l'accettazione delle connessioni in ingresso in qualsiasi momento. Al contrario, API Gateway addebita separatamente per ogni richiesta. Questo ha l'effetto che, se non arrivano richieste, non ci sono spese. In caso di carichi di traffico elevati, un servizio di Application Load Balancer di rete o un servizio di bilanciamento del carico di rete è in grado di gestire un volume maggiore di richieste a un prezzo per richiesta più economico rispetto al API Gateway. Tuttavia, se si dispone di un numero ridotto di richieste complessive o si dispone di periodi di traffico basso, il prezzo cumulativo per l'utilizzo del API Gateway dovrebbe essere più conveniente rispetto al pagamento di una tariffa oraria per mantenere un bilanciamento del carico sottoutilizzato.

API Gateway funziona utilizzando un collegamento VPC che consente ilAWSper connettersi agli host all'interno della subnet privata del VPC, utilizzando il suo indirizzo IP privato. È in grado di rilevare

questi indirizzi IP privati guardandoAWS Cloud Maprecord di individuazione dei servizi gestiti dal rilevamento del servizio Amazon ECS.

API Gateway supporta le caratteristiche seguenti.

- Terminazione SSL/TLS
- Instradamento di percorsi HTTP diversi a microservizi back-end diversi

Oltre alle funzionalità precedenti, API Gateway supporta anche l'utilizzo di autorizzatori Lambda personalizzati che è possibile utilizzare per proteggere l'API da utilizzi non autorizzati. Per ulteriori informazioni, consultaNote sul campo: API serverless basate su container con Amazon ECS e Amazon API Gateway: .

Scelta di una modalità di rete

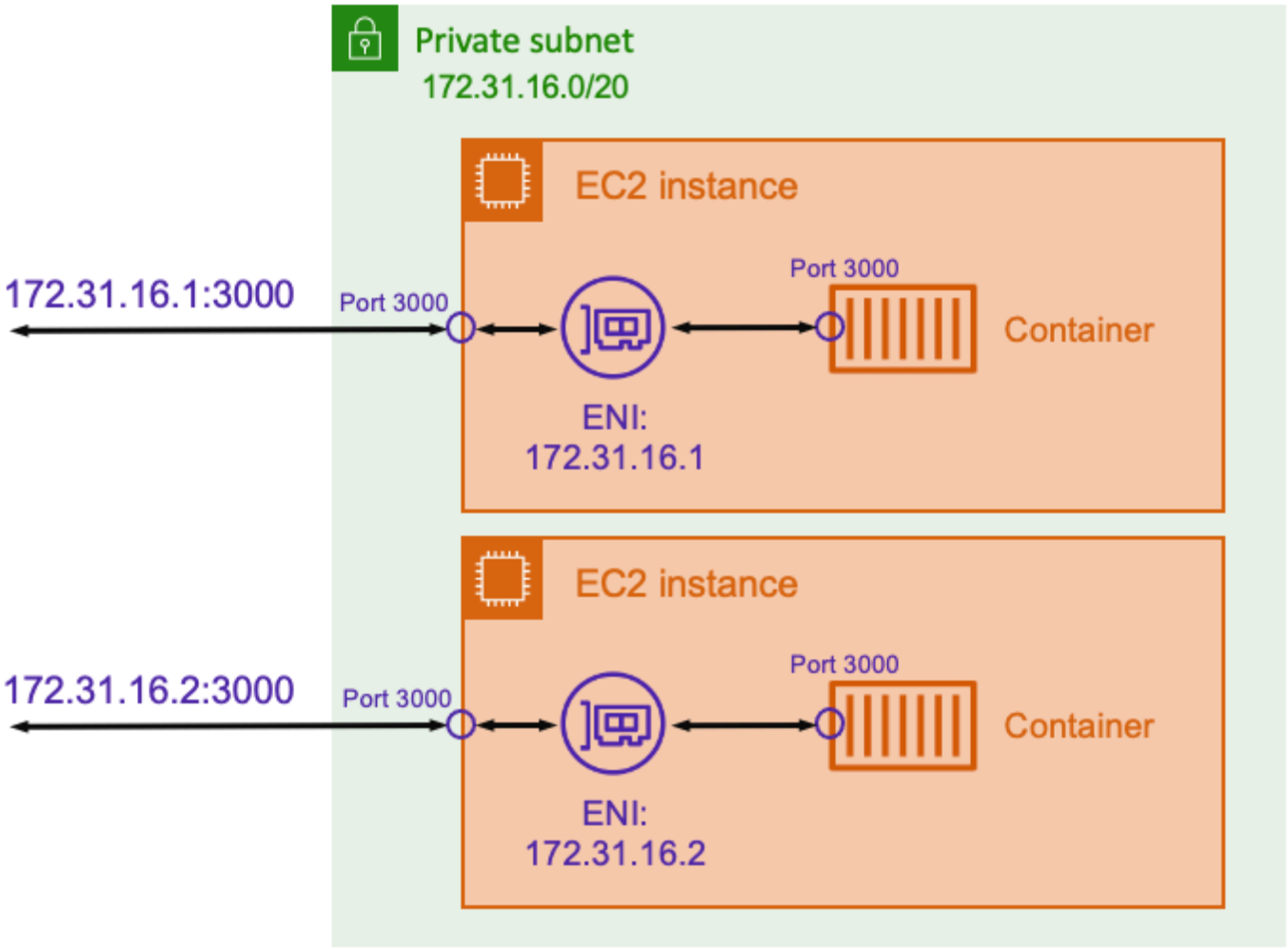
Gli approcci precedentemente menzionati per l'architettura delle connessioni di rete in ingresso e in uscita possono essere applicati a qualsiasi carico di lavoro inAWS, anche se non sono all'interno di un contenitore. Durante l'esecuzione di contenitori suAWS, è necessario prendere in considerazione un altro livello di rete. Uno dei principali vantaggi dell'utilizzo dei contenitori è che è possibile imballare più contenitori su un singolo host. Quando si esegue questa operazione, è necessario scegliere come si desidera collegare in rete i contenitori in esecuzione sullo stesso host. Di seguito sono riportate le opzioni tra cui scegliere.

Argomenti

- Modalità Host
- Modalità Bridge
- Modalità AWSVPC

Modalità Host

La`host`è la modalità di rete più semplice supportata in Amazon ECS. Utilizzando la modalità host, la rete del contenitore è collegata direttamente all'host sottostante che esegue il contenitore.



Si supponga che si sta eseguendo un contenitore Node.js con un'applicazione Express in ascolto sulla porta`3000`Simile a quello illustrato nel diagramma precedente. Quando il`host`, il contenitore riceve il traffico sulla porta 3000 utilizzando l'indirizzo IP dell'istanza host sottostante Amazon EC2. Non consigliamo di utilizzare questa modalità.

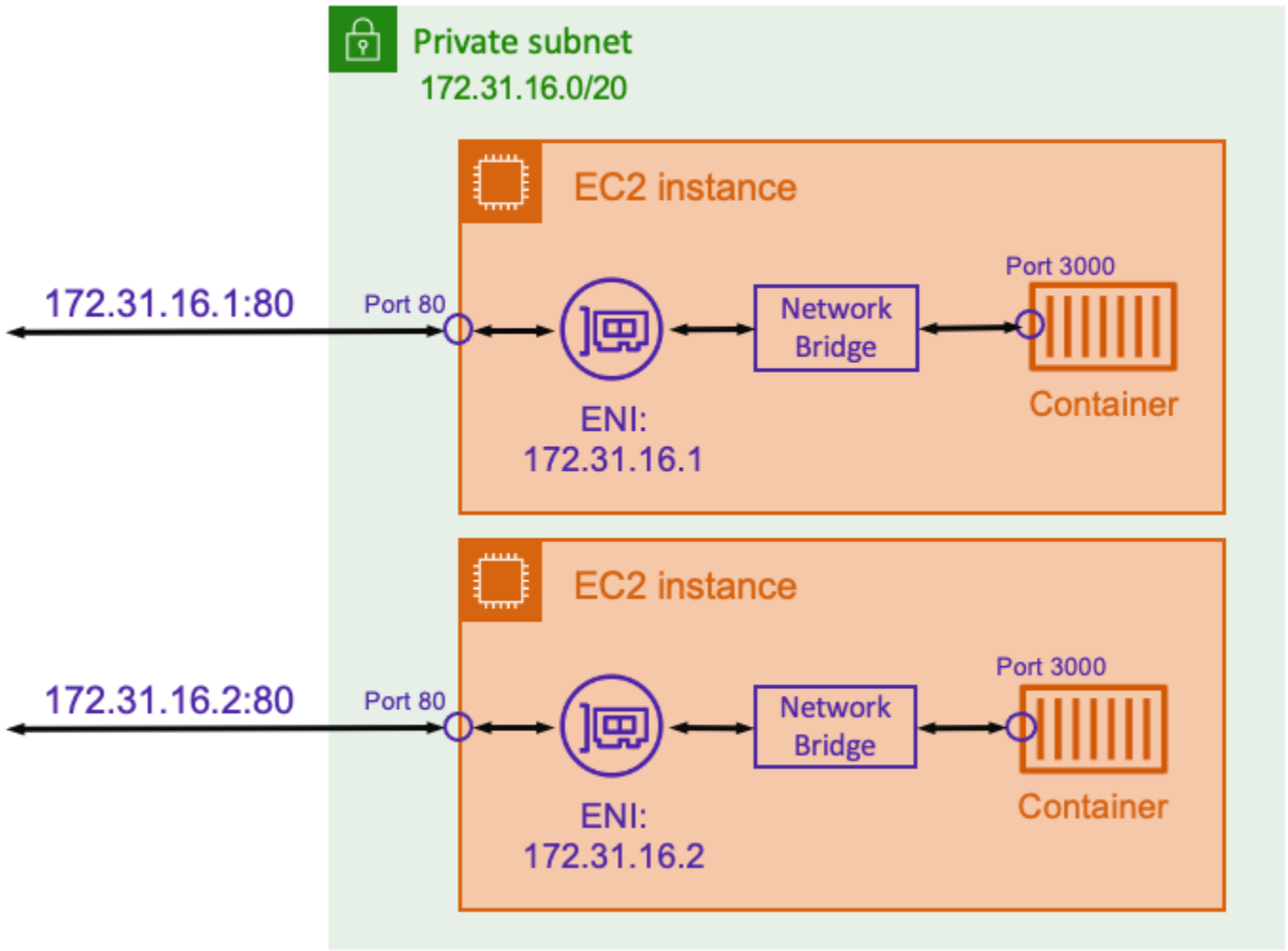
Ci sono svantaggi significativi nell'utilizzo di questa modalità di rete. Non è possibile eseguire più di una singola istanza di un'attività su ogni host. Questo perché solo la prima attività può legarsi alla sua porta richiesta sull'istanza Amazon EC2. Inoltre, non c'è modo di rimappare una porta contenitore quando utilizza`host`Modalità di rete. Ad esempio, se un'applicazione deve ascoltare un determinato numero di porta, non è possibile rimappare direttamente il numero di porta. È invece necessario gestire eventuali conflitti di porta modificando la configurazione dell'applicazione.

Ci sono anche implicazioni per la sicurezza quando si utilizza il`host`Modalità di rete. Questa modalità consente ai contenitori di rappresentare l'host e consente ai contenitori di connettersi ai servizi di rete di loopback privati sull'host.

La`host`è supportata solo per le attività Amazon ECS ospitate su istanze Amazon EC2. Non è supportato quando si utilizza Amazon ECS su Fargate.

Modalità Bridge

con`bridge`, si utilizza un bridge di rete virtuale per creare un layer tra l'host e la rete del contenitore. In questo modo, è possibile creare mapping di porte che rimappano una porta host a una porta contenitore. I mapping possono essere statici o dinamici.

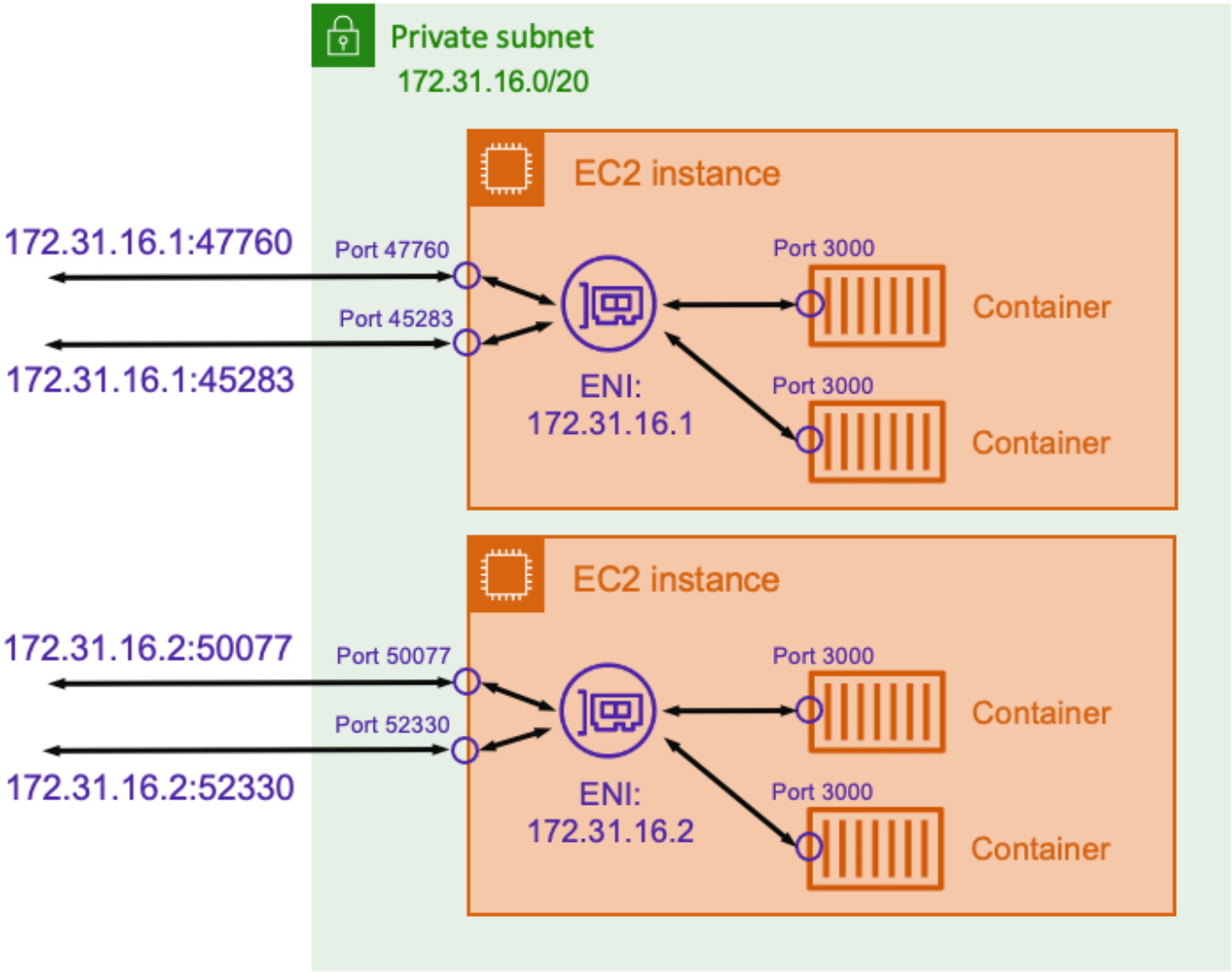


Con un mapping di porte statico, è possibile definire esplicitamente quale porta host si desidera mappare a una porta contenitore. Utilizzando l'esempio precedente, la porta`80`sull'host viene

mappato alla porta`3000`sul container. Per comunicare con l'applicazione containerizzata, si invia il traffico alla porta`80`All'indirizzo IP dell'istanza Amazon EC2. Dal punto di vista dell'applicazione containerizzata vede che il traffico in entrata sulla porta`3000`: .

Se si desidera modificare solo la porta del traffico, i mapping delle porte statiche sono adatti. Tuttavia, questo ha ancora lo stesso svantaggio dell'utilizzo del`host`Modalità di rete. Non è possibile eseguire più di una singola istanza di un'attività su ogni host. Questo perché un mapping di porta statico consente solo di mappare un singolo contenitore alla porta 80.

Per risolvere questo problema, valuta l'utilizzo di`bridge`Modalità di rete con un mapping dinamico delle porte, come illustrato nel seguente diagramma.



Non specificando una porta host nella mappatura delle porte, è possibile fare in modo che Docker scelga una porta casuale inutilizzata dall'intervallo di porte effimere e la assegni come porta host pubblica per il contenitore. Ad esempio, l'applicazione Node.js in ascolto sulla porta`3000`sul contenitore potrebbe essere assegnata una porta numero elevato casuale come`47760`sull'host Amazon EC2. Ciò significa che è possibile eseguire più copie di quel contenitore sull'host. Inoltre, ogni contenitore può essere assegnata la propria porta sull'host. Ogni copia del contenitore riceve traffico sulla porta`3000`: . Tuttavia, i client che inviano traffico a questi contenitori utilizzano le porte host assegnate in modo casuale.

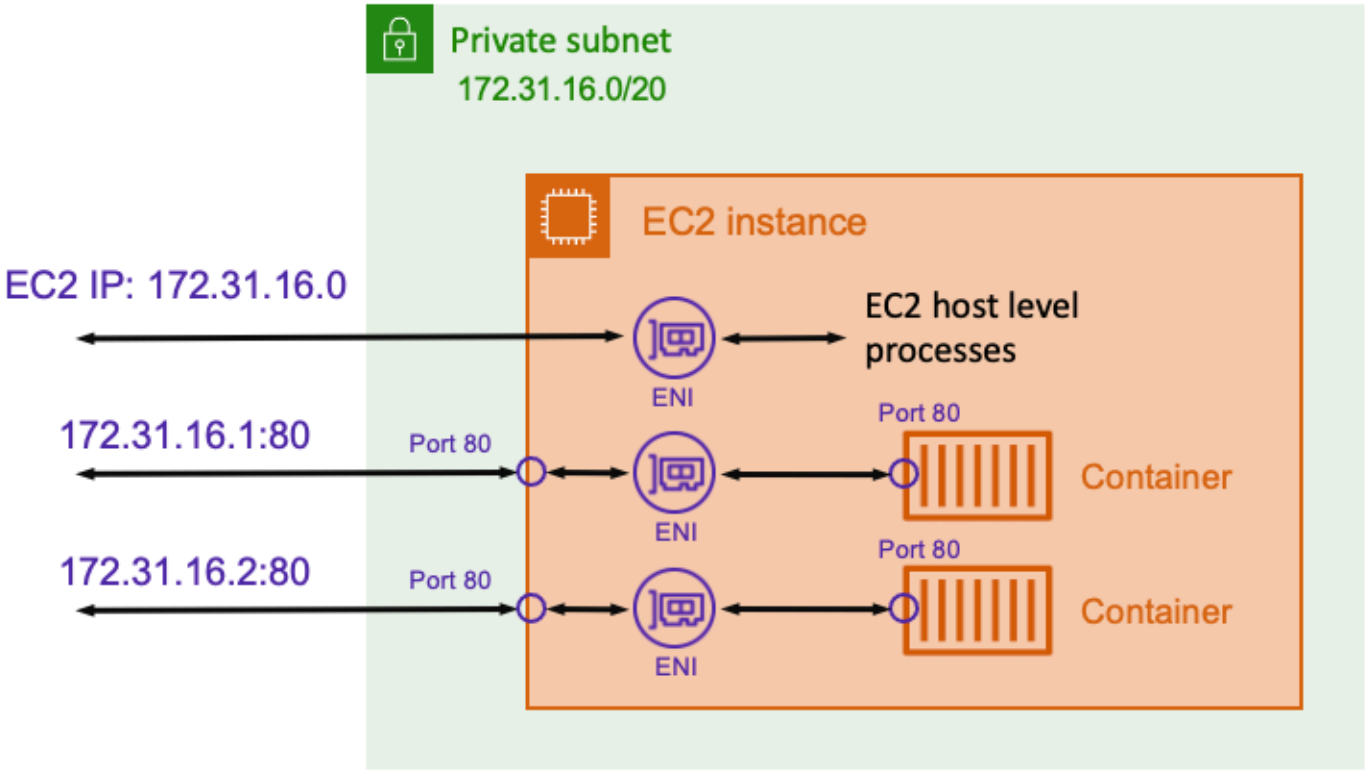
Amazon ECS ti consente di tenere traccia delle porte assegnate casualmente per ogni attività. Lo fa aggiornando automaticamente i gruppi target del bilanciamento del carico eAWS Cloud Mapper avere l'elenco degli indirizzi IP delle attività e delle porte. Ciò rende più semplice utilizzare servizi operativi utilizzando`bridge`con porte dinamiche.

Tuttavia, uno svantaggio di utilizzare il metodo`bridge`è che è difficile bloccare le comunicazioni da servizio a servizio. Poiché i servizi possono essere assegnati a qualsiasi porta casuale e inutilizzata, è necessario aprire ampi intervalli di porte tra gli host. Tuttavia, non è facile creare regole specifiche in modo che un determinato servizio possa comunicare solo a un altro servizio specifico. I servizi non dispongono di porte specifiche da utilizzare per le regole di rete dei gruppi di sicurezza.

La`bridge`è supportata solo per le attività Amazon ECS ospitate su istanze Amazon EC2. Non è supportato quando si utilizza Amazon ECS su Fargate.

Modalità AWSVPC

Con il`awsvpc`, Amazon ECS crea e gestisce un'interfaccia di rete elastica (ENI) per ogni attività e ogni attività riceve il proprio indirizzo IP privato all'interno del VPC. Questo ENI è separato dagli host sottostanti ENI. Se un'istanza Amazon EC2 esegue più attività, anche l'ENI di ciascuna attività è separata.



Nell'esempio precedente, l'istanza Amazon EC2 viene assegnata a un ENI. L'ENI rappresenta l'indirizzo IP dell'istanza EC2 utilizzata per le comunicazioni di rete a livello host. Ogni compito ha anche un ENI corrispondente e un indirizzo IP privato. Poiché ogni ENI è separato, ogni contenitore può collegarsi alla porta`80`sul compito ENI. Pertanto, non è necessario tenere traccia dei numeri di porta. Invece, è possibile inviare il traffico alla porta`80`All'indirizzo IP dell'attività ENI.

Il vantaggio di utilizzare il`awsvpc`è che ogni attività ha un gruppo di sicurezza separato per consentire o negare il traffico. Ciò significa una maggiore flessibilità per controllare le comunicazioni tra attività e servizi a un livello più granulare. È inoltre possibile configurare un'attività per negare il traffico in ingresso da un'altra attività situata sullo stesso host.

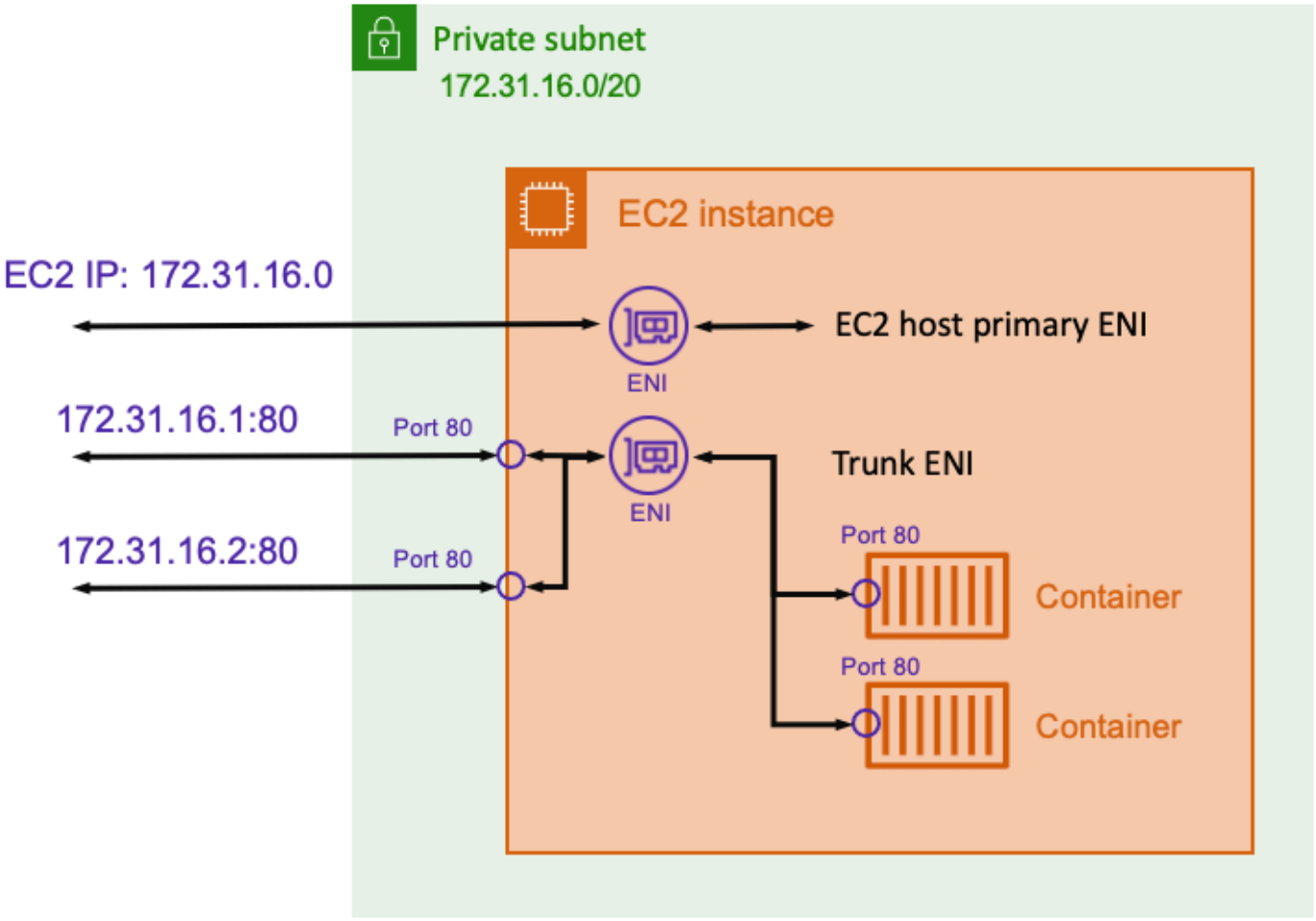
La`awsvpc`è supportata per le attività Amazon ECS ospitate sia su Amazon EC2 che su Fargate. Tieni presente che, quando usi Fargate, il`awsvpc`è necessaria la modalità di rete.

Quando utilizzi il comando`awsvpc`ci sono alcune sfide di cui dovresti essere consapevole.

Aumento della densità delle attività con ENI Trunking

Il più grande svantaggio dell'utilizzo di`awsvpc`Con attività ospitate su istanze Amazon EC2 è che le istanze EC2 hanno un limite al numero di ENI che possono essere collegate. Questo limita il numero

di attività che è possibile posizionare su ogni istanza. Amazon ECS fornisce la funzione ENI trunking che aumenta il numero di ENI disponibili per ottenere una maggiore densità di attività.



Quando si utilizza ENI trunking, vengono utilizzati per impostazione predefinita due allegati ENI. Il primo è l'ENI primario dell'istanza, che viene utilizzato per qualsiasi processo a livello host. Il secondo è il tronco ENI, che Amazon ECS crea. Questa funzionalità è supportata solo su tipi di istanza Amazon EC2 specifici.

Considerate questo esempio. Senza trunking ENI, un`c5.large`che dispone di due vCPUs può ospitare solo due attività. Tuttavia, con il trunking ENI, un`c5.large`che ha due vCPU può ospitare fino a dieci attività. Ogni attività ha un indirizzo IP e un gruppo di sicurezza diversi. Per ulteriori informazioni sui tipi di istanza disponibili e sulla relativa densità, consultaTipi di istanze Amazon EC2 supportatenellaAmazon Elastic Container Service: .

Il trunking ENI non ha alcun impatto sulle prestazioni di runtime in termini di latenza o larghezza di banda. Tuttavia, aumenta il tempo di avvio delle attività. È necessario assicurarsi che, se si utilizza il

trunking ENI, le regole di scalabilità automatica e gli altri carichi di lavoro che dipendono dal tempo di avvio delle attività continuino a funzionare come previsto.

Per ulteriori informazioni, consultaTrunking dell'interfaccia di rete elasticanellaAmazon Elastic Container Service: .

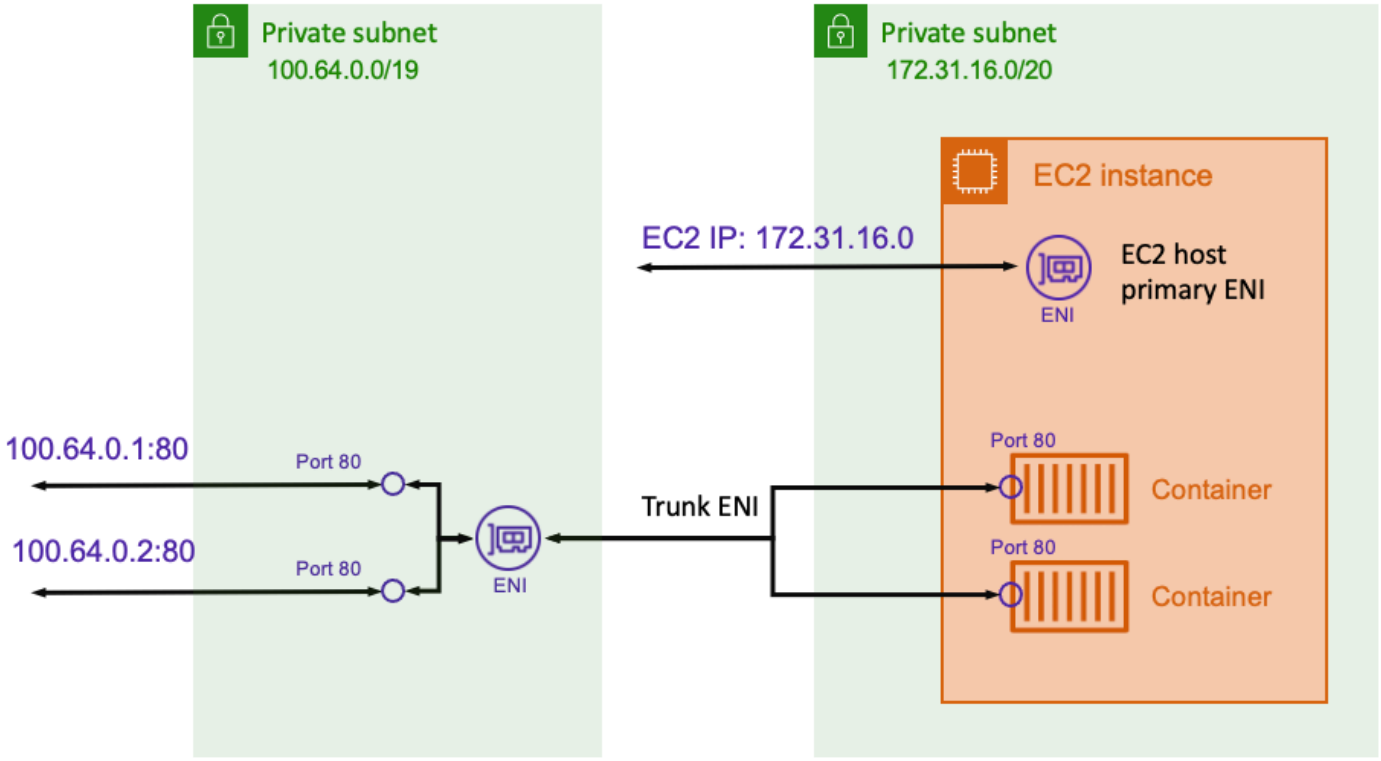
Prevenire l'esaurimento degli indirizzi IP

Assegnando un indirizzo IP separato a ciascuna attività, è possibile semplificare l'infrastruttura complessiva e gestire gruppi di sicurezza che offrono un elevato livello di sicurezza. Tuttavia, questa configurazione può portare all'esaurimento IP.

Il VPC predefinito inAWSdispone di subnet pre-provisioning che dispongono di un`/20`Linea CIDR. Ciò significa che ogni subnet dispone di 4.091 indirizzi IP disponibili. Si noti che diversi indirizzi IP all'interno del`/20`sono riservati per l'utilizzo specifico di AWS. Considerate questo esempio. Puoi distribuire le applicazioni tra tre sottoreti in tre zone di disponibilità per elevata disponibilità. In questo caso, è possibile utilizzare circa 12.000 indirizzi IP nelle tre subnet.

Utilizzando ENI trunking, ogni istanza Amazon EC2 che avvii richiede due indirizzi IP. Un indirizzo IP viene utilizzato per l'ENI primario e l'altro indirizzo IP viene utilizzato per il trunk ENI. Ogni attività Amazon ECS sull'istanza richiede un indirizzo IP. Se si avvia un carico di lavoro estremamente grande, è possibile esaurire gli indirizzi IP disponibili. Ciò potrebbe comportare errori di lancio di Amazon EC2 o errori di avvio delle attività. Questi errori si verificano perché gli ENI non possono aggiungere indirizzi IP all'interno del VPC se non ci sono indirizzi IP disponibili.

Quando utilizzi il comando`awsvpc`, è necessario valutare i requisiti dell'indirizzo IP e assicurarsi che gli intervalli CIDR della subnet soddisfino le proprie esigenze. Se è già stato iniziato a utilizzare un VPC con subnet di piccole dimensioni e inizia a esaurire lo spazio degli indirizzi, è possibile aggiungere una subnet secondaria.



Utilizzando ENI trunking, il CNI di Amazon VPC può essere configurato per utilizzare ENI in uno spazio di indirizzi IP diverso rispetto all'host. In questo modo, puoi assegnare al tuo host Amazon EC2 e alle tue attività intervalli di indirizzi IP diversi che non si sovrappongono. Nel diagramma di esempio, l'indirizzo IP dell'host EC2 si trova in una subnet con il`172.31.16.0/20`Intervallo IP. Tuttavia, le attività in esecuzione sull'host vengono assegnati indirizzi IP nel`100.64.0.0/19`Intervallo. Utilizzando due intervalli IP indipendenti, non è necessario preoccuparsi di attività che consumano troppi indirizzi IP e non lasciano indirizzi IP sufficienti per le istanze.

Utilizzo della modalità dual stack IPv6

La`awsvpc`è compatibile con i VPC configurati per la modalità dual stack IPv6. Un VPC che utilizza la modalità dual stack può comunicare via IPv4, IPv6 o entrambi. Ogni subnet del VPC può avere sia un intervallo CIDR IPv4 che un intervallo CIDR IPv6. Per ulteriori informazioni, consultaIndirizzamento IP nel VPCnellaGuida per l'utente di Amazon VPC: .

Non è possibile disabilitare il supporto IPv4 per il VPC e le sottoreti in modo da risolvere i problemi di esaurimento IPv4. Tuttavia, con il supporto IPv6 puoi utilizzare alcune nuove funzionalità, in particolare il gateway Internet egress-only. Un gateway Internet di sola uscita consente alle attività

di utilizzare il proprio indirizzo IPv6 instradabile pubblicamente per avviare connessioni in uscita a Internet. Ma il gateway Internet egress-only non consente connessioni da Internet. Per ulteriori informazioni, consultaInternet Gateway egress-onlynellaGuida per l'utente di Amazon VPC: .

Connessione adAWSservizi dall'interno del tuo VPC

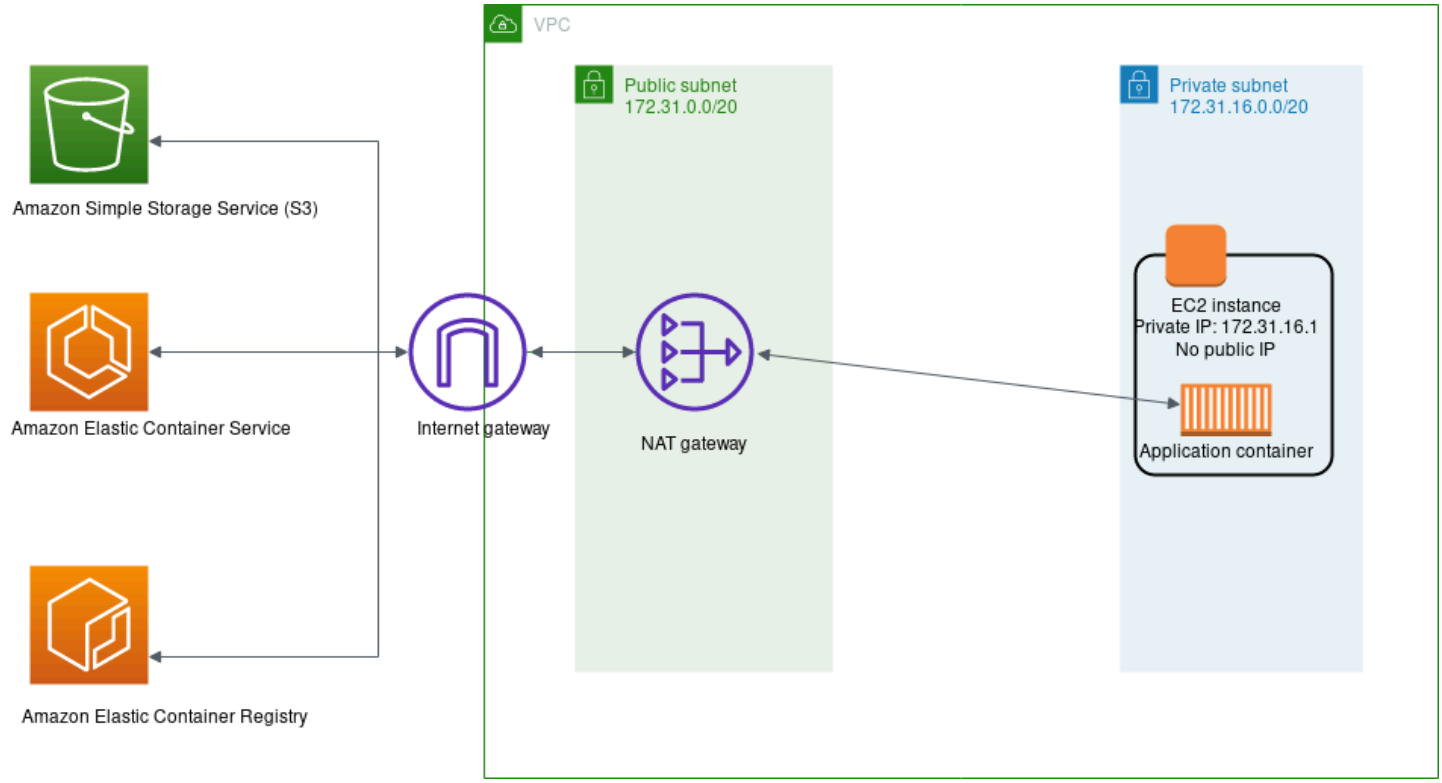
Affinché Amazon ECS funzioni correttamente, l'agente contenitore ECS in esecuzione su ciascun host deve comunicare con il piano di controllo Amazon ECS. Se stai memorizzando le immagini del contenitore in Amazon ECR, gli host Amazon EC2 devono comunicare all'endpoint del servizio Amazon ECR e ad Amazon S3, dove sono archiviati i layer immagine. Se usi altriAWSper l'applicazione containerizzata, ad esempio i dati persistenti archiviati in DynamoDB, verificare che questi servizi dispongano anche del supporto di rete necessario.

Argomenti

- Gateway NAT
- AWS PrivateLink

Gateway NAT

L'utilizzo di un gateway NAT è il modo più semplice per garantire che le tue attività Amazon ECS possano accedere ad altriAWSServizi . Per ulteriori informazioni su questo approccio, consultaUtilizzo di una subnet privata e di un gateway NAT: .



Di seguito sono riportati gli svantaggi dell'utilizzo di questo approccio:

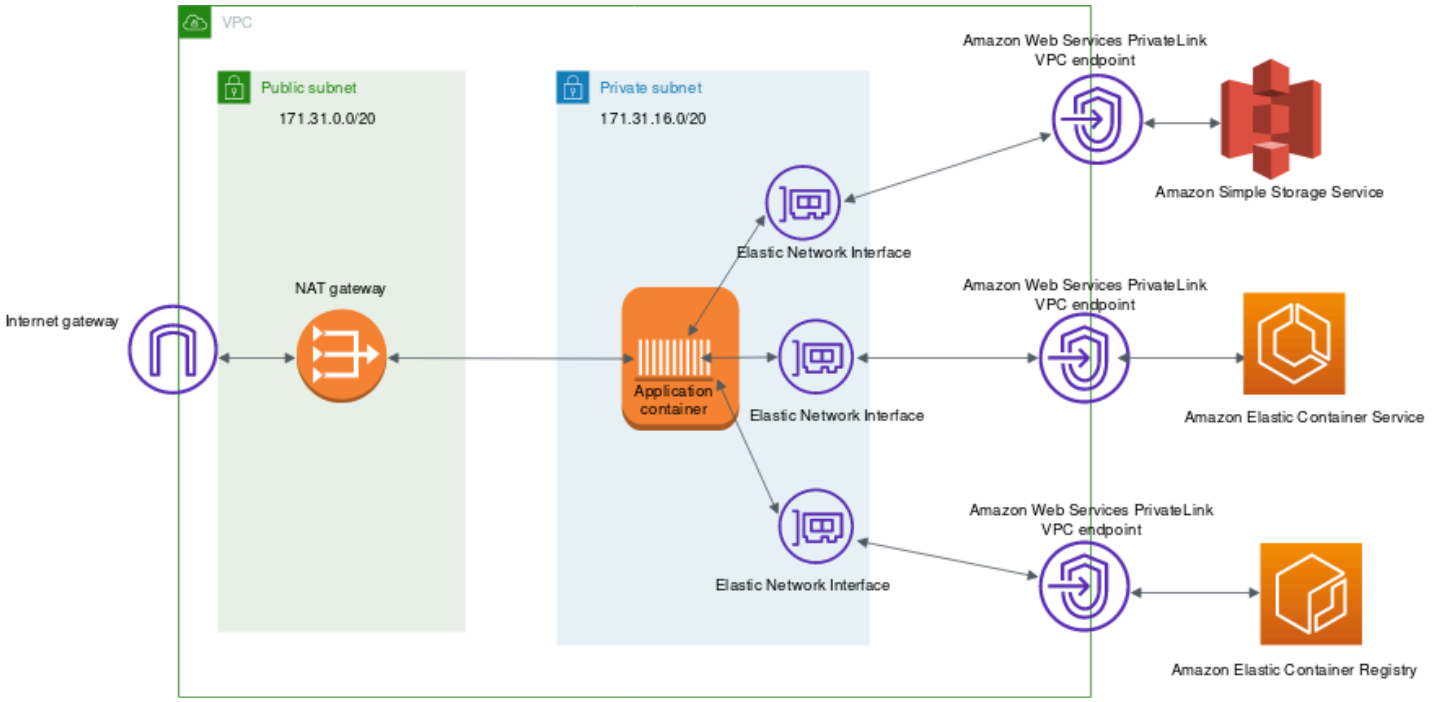
- Non è possibile limitare le destinazioni con cui il gateway NAT può comunicare. Inoltre, non puoi limitare le destinazioni a cui il tuo pneumatico back-end può comunicare senza interrompere tutte le comunicazioni in uscita dal tuo VPC.
- I gateway NAT addebitano ogni GB di dati che passa attraverso. Se utilizzi il gateway NAT per scaricare file di grandi dimensioni da Amazon S3 o esegui un volume elevato di query di database su DynamoDB, ti verrà addebitato ogni GB di larghezza di banda. Inoltre, i gateway NAT supportano 5 Gbps di larghezza di banda e scalano automaticamente fino a 45 Gbps. Se si instradano attraverso un singolo gateway NAT, le applicazioni che richiedono connessioni con larghezza di banda molto elevata potrebbero incontrare vincoli di rete. Come soluzione alternativa, è possibile dividere il carico di lavoro tra più subnet e assegnare a ciascuna subnet il proprio gateway NAT.

AWS PrivateLink

AWS PrivateLinkfornisce connettività privata tra VPC,AWSe le reti locali senza esporre il traffico a Internet pubblico.

Una delle tecnologie utilizzate per ottenere questo risultato è l'endpoint VPC. Un endpoint VPC consente connessioni private tra il VPC e il supporto supportatoAWSe servizi endpoint VPC. Il traffico tra il VPC e gli altri servizi non lascia la rete Amazon. Un endpoint VPC non richiede un gateway Internet, un gateway privato virtuale, un dispositivo NAT, una connessione VPN oAWS Direct ConnectConnessione. Le istanze Amazon EC2 nel VPC non richiedono indirizzi IP pubblici per comunicare con risorse nel servizio.

Il diagramma riportato di seguito illustra come la comunicazione aAWSfunziona quando si utilizzano endpoint VPC invece di un gateway Internet.AWS PrivateLinkpredispone interfacce di rete elastiche (ENI) all'interno della subnet, e le regole di routing VPC vengono utilizzate per inviare qualsiasi comunicazione al nome host del servizio attraverso l'ENI, direttamente alla destinazioneAWSServizio. Questo traffico non deve più utilizzare il gateway NAT o il gateway Internet.



Di seguito sono riportati alcuni degli endpoint VPC comuni utilizzati con il servizio Amazon ECS.

- Endpoint VPC del gateway S
- Endpoint VPC di DynamoDB
- Endpoint VPC di Amazon ECS
- Endpoint VPC di Amazon ECR

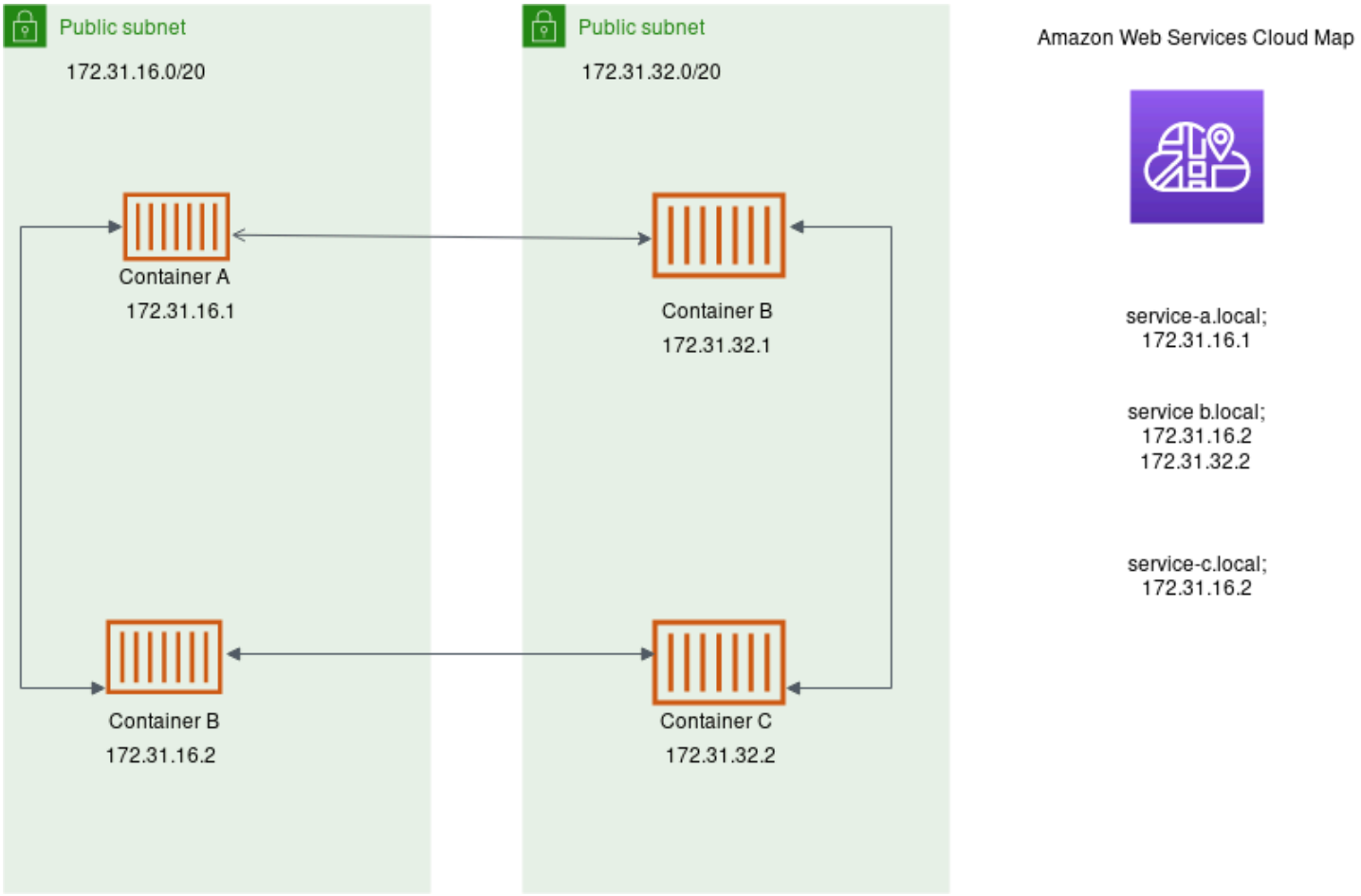
Molti altriAWSSupportano gli endpoint VPC. Se fai un uso pesante di qualsiasiAWS, è necessario cercare la documentazione specifica per tale servizio e come creare un endpoint VPC per tale traffico.

Networking tra i servizi Amazon ECS in un VPC

Utilizzando i contenitori Amazon ECS in un VPC, è possibile versare applicazioni monolitiche in parti separate che possono essere distribuite e scalate in modo indipendente in un ambiente sicuro. Tuttavia, può essere difficile assicurarsi che tutte queste parti, sia all'interno che all'esterno di un VPC, possano comunicare tra loro. Esistono diversi approcci per facilitare la comunicazione, tutti con diversi vantaggi e svantaggi.

Utilizzo dell'individuazione dei servizi

Un approccio per la comunicazione da servizio a servizio è la comunicazione diretta utilizzando l'individuazione dei servizi. In questo approccio puoi utilizzare l'AWS Cloud Mapintegrazione con Amazon ECS. Utilizzando l'individuazione dei servizi, Amazon ECS sincronizza l'elenco delle attività avviate inAWS Cloud Map, che mantiene un nome host DNS che risolve gli indirizzi IP interni di una o più attività di quel particolare servizio. Altri servizi in Amazon VPC possono utilizzare questo nome host DNS per inviare il traffico direttamente a un altro contenitore utilizzando il suo indirizzo IP interno. Per ulteriori informazioni, consultaIdentificazione dei servizinellaAmazon Elastic Container Service: .



Nel diagramma precedente, ci sono tre servizi.`serviceA`ha un contenitore e comunica con`serviceB`, che ha due container.`serviceB`deve anche comunicare con`serviceC`, che ha un contenitore. Ogni contenitore in tutti e tre questi servizi può utilizzare i nomi DNS interni daAWS Cloud Mapper trovare gli indirizzi IP interni di un contenitore dal servizio downstream a cui deve comunicare.

Questo approccio alla comunicazione service-to-service fornisce bassa latenza. A prima vista, è anche semplice in quanto non ci sono componenti aggiuntivi tra i contenitori. Il traffico viaggia direttamente da un contenitore all'altro contenitore.

Questo approccio è adatto quando si utilizza il metodo`awsvpc`modalità di rete, in cui ogni attività ha il proprio indirizzo IP univoco. La maggior parte del software supporta solo l'uso di DNSA, che si risolvono direttamente agli indirizzi IP. Quando utilizzi il comando`awsvpc`, l'indirizzo IP per ogni attività è unARecord. Tuttavia, se utilizzi`bridge`, più contenitori potrebbero condividere lo stesso indirizzo IP. Inoltre, i mapping delle porte dinamiche causano ai contenitori di essere assegnati in modo casuale numeri di porta su quel singolo indirizzo IP. A questo punto, unAnon è più sufficiente per l'individuazione del servizio. È inoltre necessario utilizzare un`SRV`Record. Questo tipo di

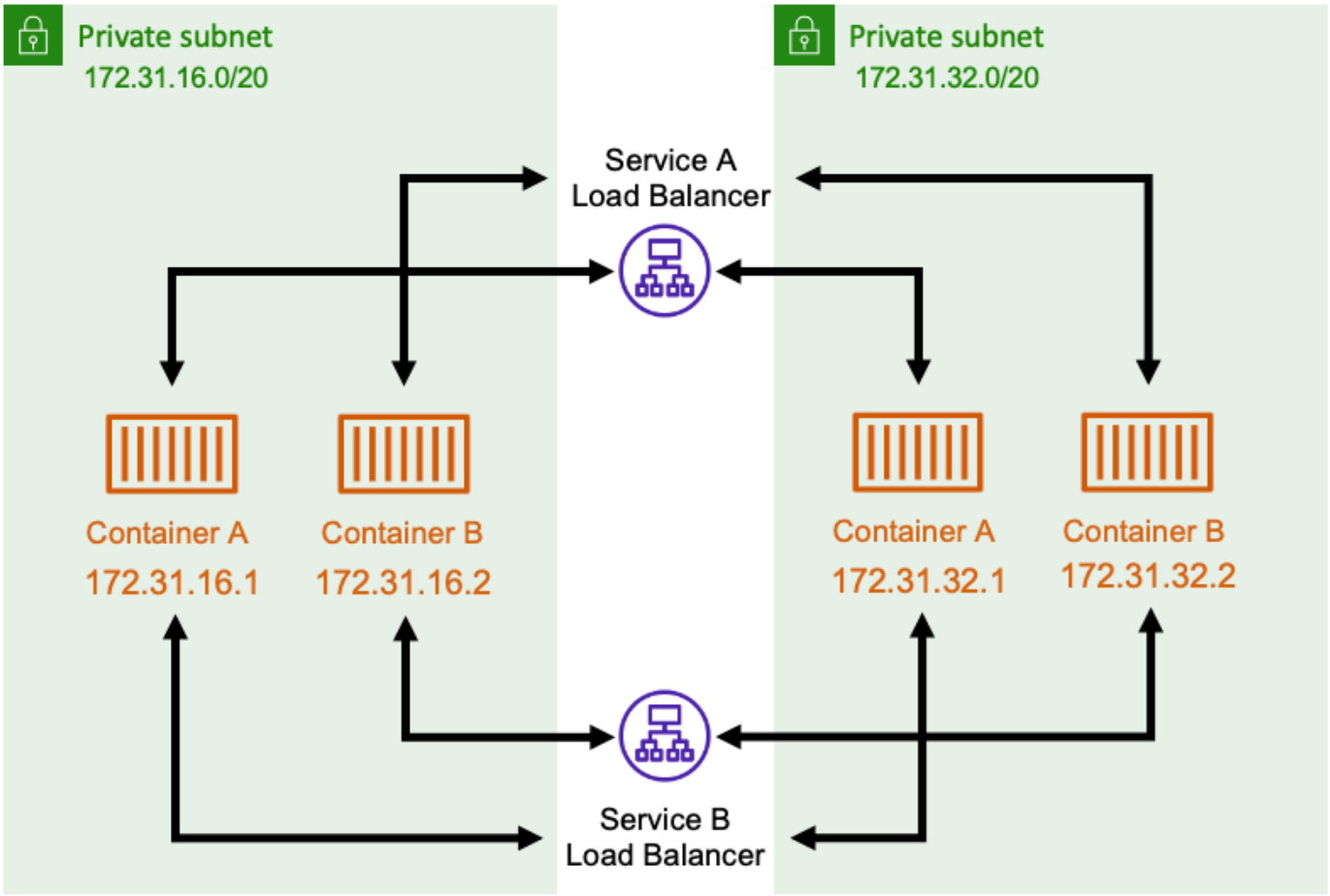
record può tenere traccia degli indirizzi IP e dei numeri di porta, ma richiede che le applicazioni siano configurate in modo appropriato. Alcune applicazioni precompilate utilizzate potrebbero non supportareSRVRecord.

Un altro vantaggio dellaawsvpcè che si dispone di un gruppo di sicurezza univoco per ogni servizio. È possibile configurare questo gruppo di protezione per consentire le connessioni in ingresso solo dai servizi upstream specifici che devono comunicare con tale servizio.

Lo svantaggio principale della comunicazione diretta da servizio a servizio utilizzando l'individuazione dei servizi è che è necessario implementare una logica aggiuntiva per avere tentativi e gestire gli errori di connessione. I record DNS hanno un periodo TTL (Time-To-Live) che controlla per quanto tempo vengono memorizzati nella cache. Ci vuole del tempo per l'aggiornamento del record DNS e per la scadenza della cache in modo che le applicazioni possano prendere la versione più recente del record DNS. Quindi, l'applicazione potrebbe finire per risolvere il record DNS per puntare a un altro contenitore che non è più lì. La tua applicazione deve gestire i tentativi e avere la logica per ignorare i backend errati.

Utilizzo di un sistema di bilanciamento del carico interno

Un altro approccio alla comunicazione service-to-service consiste nell'utilizzare un bilanciamento del carico interno. Un bilanciamento del carico interno esiste interamente all'interno del VPC ed è accessibile solo ai servizi all'interno del VPC.



Il servizio di bilanciamento del carico mantiene la disponibilità elevata distribuendo risorse ridondanti in ogni subnet. Quando un contenitore da`serviceA`ha bisogno di comunicare con un contenitore da`serviceB`, apre una connessione al sistema di bilanciamento del carico. Il bilanciamento del carico apre quindi una connessione a un contenitore da`service B`: . Il bilanciamento del carico funge da luogo centralizzato per la gestione di tutte le connessioni tra ciascun servizio.

Se un contenitore da`serviceB`si arresta, quindi il bilanciamento del carico può rimuovere il contenitore dal pool. Il bilanciamento del carico esegue anche controlli di integrità per ogni destinazione a valle nel proprio pool e può rimuovere automaticamente gli obiettivi non validi dal pool fino a quando non diventano nuovamente integri. Le applicazioni non devono più essere consapevoli di quanti contenitori a valle ci sono. Aprono solo le connessioni al sistema di bilanciamento del carico.

Questo approccio è vantaggioso per tutte le modalità di rete. Il bilanciamento del carico è in grado di tenere traccia degli indirizzi IP delle attività quando si utilizza il`awsvpc`modalità di rete, così come combinazioni più avanzate di indirizzo IP e porta quando si utilizza il`bridge`Modalità di rete.

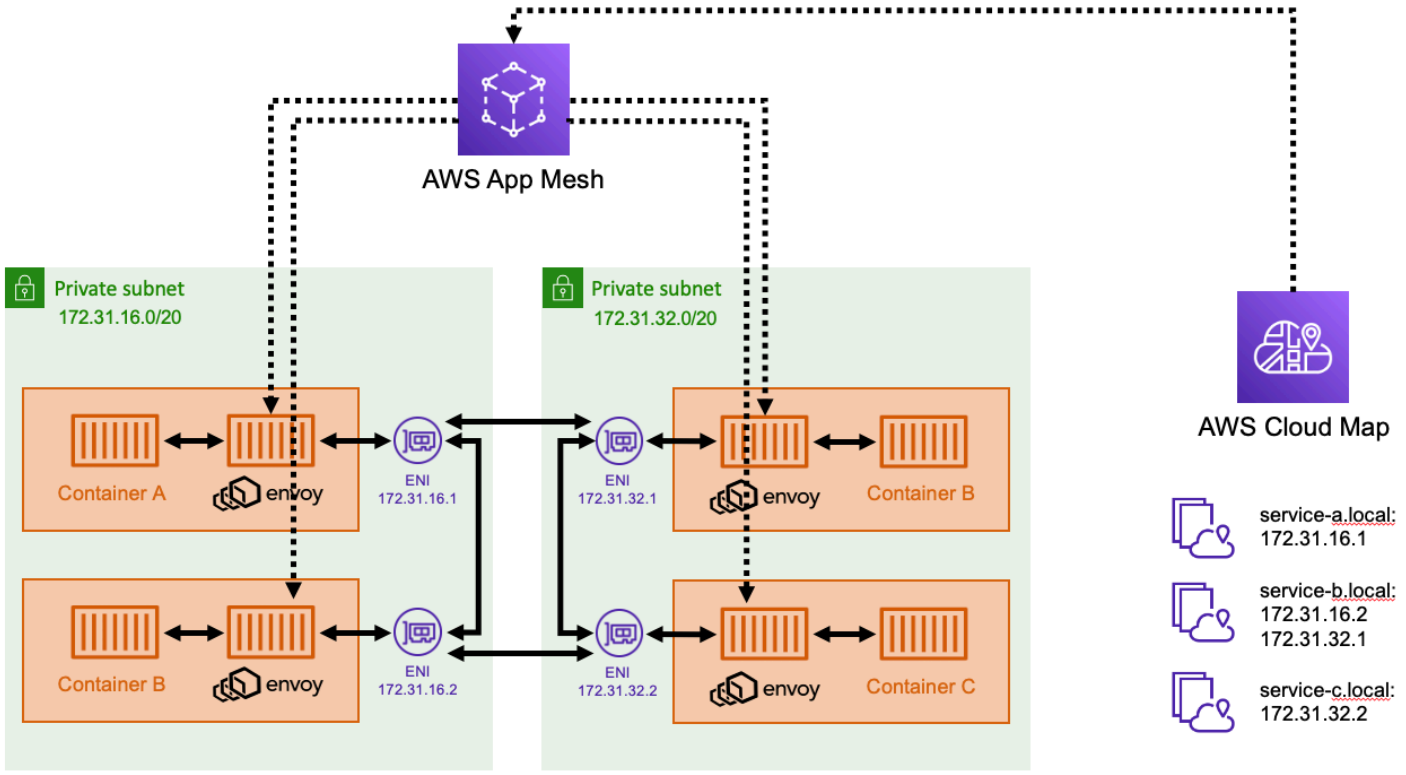
Distribuisce uniformemente il traffico su tutte le combinazioni di indirizzi IP e porte, anche se diversi contenitori sono effettivamente ospitati sulla stessa istanza Amazon EC2, solo su porte diverse.

L'unico svantaggio di questo approccio è il costo. Per essere a disponibilità elevata, il servizio di bilanciamento del carico deve disporre di risorse in ogni zona di disponibilità. Ciò aggiunge un costo aggiuntivo a causa del sovraccarico di pagamento per il bilanciamento del carico e per la quantità di traffico che passa attraverso il bilanciamento del carico.

Tuttavia, è possibile ridurre i costi generali facendo in modo che più servizi condividano un servizio di bilanciamento del carico. Ciò è particolarmente adatto per i servizi REST che utilizzano un servizio di Application Load Balancer. È possibile creare regole di routing basate su percorsi che instradano il traffico verso servizi diversi. Ad esempio:`/api/user/*`potrebbe instradare a un contenitore che fa parte del`user`Servizio, mentre`/api/order/*`potrebbe instradare verso l'associato`order`Servizio. Con questo approccio, si paga solo per un Application Load Balancer e si dispone di un URL coerente per l'API. Tuttavia, è possibile dividere il traffico in vari microservizi sul back-end.

Utilizzo di una mesh dei servizi

AWS App Meshè una mesh di servizi che consente di gestire un gran numero di servizi e di controllare meglio il modo in cui il traffico viene instradato tra i servizi. App Mesh funziona come intermediario tra l'individuazione dei servizi di base e il bilanciamento del carico. Con App Mesh, le applicazioni non interagiscono direttamente tra loro, ma non utilizzano nemmeno un bilanciamento del carico centralizzato. Invece, ogni copia del tuo compito è accompagnata da un proxy sidecar inviato. Per ulteriori informazioni, consulta Cos'è AWS App Mesh? nella Guida per l'utente di AWS App Mesh.



Nel diagramma precedente, ogni attività ha un servizio proxy dell'inviato. Questo sidecar è responsabile dell'inoltro di tutto il traffico in entrata e in uscita per l'attività. Il piano di controllo App Mesh utilizzaAWS Cloud Mapper ottenere l'elenco dei servizi disponibili e gli indirizzi IP di attività specifiche. Quindi App Mesh fornisce la configurazione al sidecar proxy Inviato. Questa configurazione include l'elenco dei contenitori disponibili a cui è possibile connettersi. L'Envoy proxy sidecar effettua anche controlli sanitari contro ogni bersaglio per assicurarsi che siano disponibili.

Questo approccio fornisce le funzionalità di individuazione dei servizi, con la facilità del servizio di bilanciamento del carico gestito. Le applicazioni non implementano la stessa logica di bilanciamento del carico all'interno del loro codice perché il sidecar proxy Envoy gestisce il bilanciamento del carico. Il proxy Inviato può essere configurato per rilevare errori e riprovare le richieste non riuscite. Inoltre, può anche essere configurato per utilizzare MTL per crittografare il traffico in transito e assicurarsi che le applicazioni stiano comunicando a una destinazione verificata.

Esistono poche differenze tra un proxy Inviato e un servizio di bilanciamento del carico. In breve, con Envoy proxy, sei responsabile della distribuzione e della gestione del tuo sidecar proxy Envoy. La sidecar proxy inviato utilizza parte della CPU e della memoria che allocate all'attività Amazon ECS. Ciò aggiunge un sovraccarico al consumo delle risorse delle attività e un carico di lavoro operativo aggiuntivo per mantenere e aggiornare il proxy quando necessario.

App Mesh e un proxy Inviato consentono una latenza estremamente bassa tra le attività. Questo perché il proxy Inviato viene eseguito collocato in ogni attività. C'è solo un'istanza per il salto di rete, tra un proxy Inviato e un altro proxy Inviato. Ciò significa che c'è anche meno sovraccarico di rete rispetto a quando si utilizzano i bilanciatori di carico. Quando si utilizzano i bilanciatori di carico, ci sono due salti di rete. Il primo è dall'attività a monte al servizio di bilanciamento del carico e il secondo è dal bilanciamento del carico all'attività a valle.

Servizi di rete traAWSaccount e VPC

Se fai parte di un'organizzazione con più team e divisioni, probabilmente distribuisci i servizi in modo indipendente in vPC separati all'interno di unAWSo in VPC associati a più singoliAWSaccount. Indipendentemente dal modo in cui si distribuiscono i servizi, si consiglia di integrare i componenti di rete per facilitare l'instradamento del traffico tra VPC. Per questo, diversiAWSpossono essere utilizzati per integrare i componenti di rete esistenti.

- AWS Transit Gateway — È necessario considerare prima questo servizio di rete. Questo servizio funge da hub centrale per il routing delle connessioni tra vPC Amazon,AWSe le reti locali. Per ulteriori informazioni, consultaChe cos'è un gateway di transito?nellaGuida ai gateway di transito VPC di Amazon: .
- Supporto VPC e VPN Amazon: è possibile utilizzare questo servizio per creare connessioni VPN da sito a sito per la connessione di reti locali al VPC. Per ulteriori informazioni, consulta Che cos'è AWS Site-to-Site VPN? nella Guida per l'utente AWS Site-to-Site VPN.
- Amazon VPC: puoi utilizzare il peering Amazon VPC per aiutarti a connettere più VPC, nello stesso account o tra più account. Per ulteriori informazioni, consulta Che cos'è il peering di VPC? nella Amazon VPC Peering Guide.
- VPC condivisi: è possibile utilizzare subnet VPC e VPC in piùAWSaccount. Per ulteriori informazioni, consultaUtilizzo dei VPC condivisinellaGuida per l'utente di Amazon VPC: .

Ottimizzazione e risoluzione dei problemi

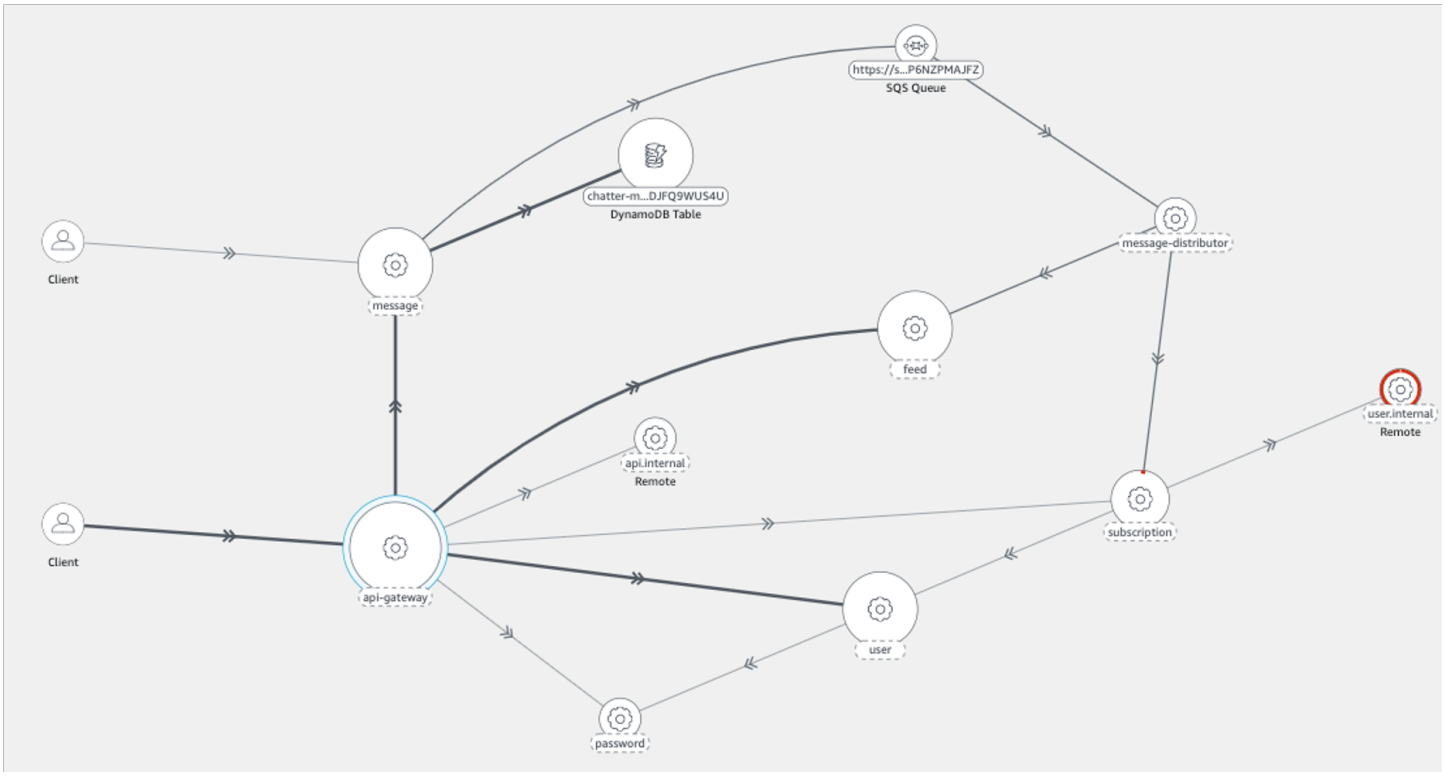
I servizi e le funzionalità seguenti consentono di ottenere informazioni dettagliate sulle configurazioni della rete e dei servizi. Puoi utilizzare queste informazioni per risolvere i problemi di rete e ottimizzare i servizi.

Informazioni sulle informazioni sui container CloudWatch

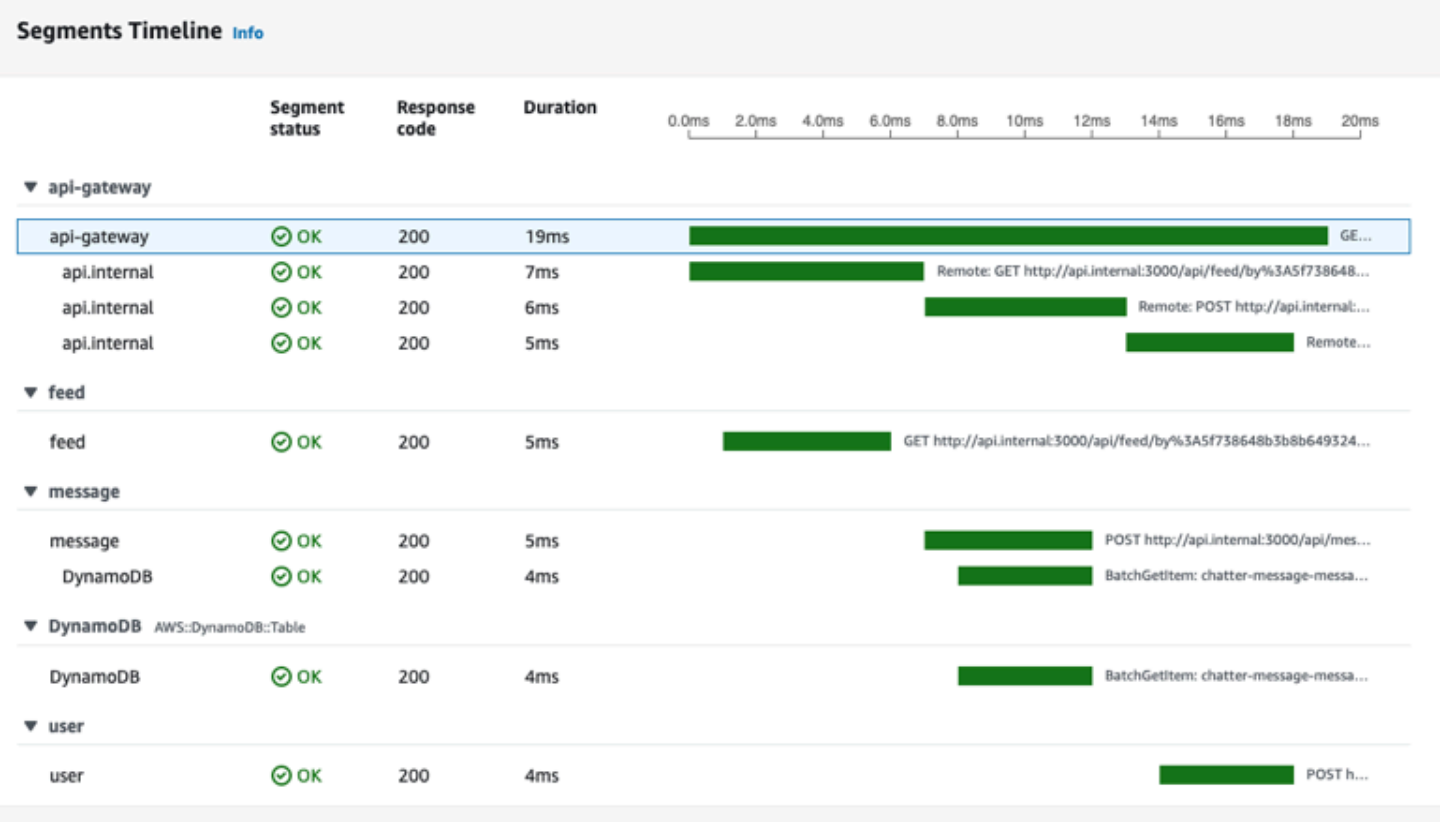
CloudWatch Container Insights raccoglie, aggrega e riepiloga parametri e log di applicazioni e microservizi containerizzati. Le metriche includono l'utilizzo di risorse come CPU, memoria, dischi e rete. Sono disponibili nei dashboard automatici CloudWatch. Per ulteriori informazioni, consultaImpostazione di informazioni sui contenitori su Amazon ECSnellaGuida per l'utente di Amazon CloudWatch: .

AWS X-Ray

AWS X-Rayè un servizio di analisi che è possibile utilizzare per raccogliere informazioni sulle richieste di rete effettuate dall'applicazione. È possibile utilizzare l'SDK per strumentare l'applicazione e acquisire tempi e codici di risposta del traffico tra i servizi e tra i servizi eAWSEndpoint del servizio. Per ulteriori informazioni, consultaChe cos'èAWS X-RaynellaAWS X-RayGuida per gli sviluppatori: .



Puoi anche esplorareAWS X-Raygrafici di come i vostri servizi di rete tra loro. In alternativa, utilizzali per esplorare statistiche aggregate sulle prestazioni di ciascun collegamento da servizio a servizio. Infine, è possibile approfondire qualsiasi transazione specifica per vedere come i segmenti che rappresentano le chiamate di rete sono associati a quella particolare transazione.



È possibile utilizzare queste funzionalità per identificare se esiste un collo di bottiglia di rete o se un servizio specifico all'interno della rete non funziona come previsto.

Log di flusso VPC

Puoi utilizzare i log di flusso di Amazon VPC per analizzare le prestazioni della rete e eseguire il debug dei problemi di connettività. Con i registri di flusso VPC abilitati, è possibile acquisire un registro di tutte le connessioni nel VPC. Questi includono connessioni alle interfacce di rete associate a Elastic Load Balancing, Amazon RDS, gateway NAT e altre chiaviAWSche potresti utilizzare. Per ulteriori informazioni, consulta l'argomento relativo ai Log di flusso VPC nella Guida per l'utente di Amazon VPC.

Consigli di ottimizzazione della rete

Ci sono alcune impostazioni che è possibile ottimizzare per migliorare la rete.

Ulimit di nofile

Se si prevede che l'applicazione abbia un traffico elevato e gestisca molte connessioni simultanee, è necessario prendere in considerazione la quota di sistema per il numero di file consentiti. Quando

ci sono molti socket di rete aperti, ognuno deve essere rappresentato da un descrittore di file. Se la quota del descrittore di file è troppo bassa, limiterà i socket di rete,. Ciò si traduce in connessioni o errori non riusciti. Puoi aggiornare la quota specifica del contenitore per il numero di file nella definizione dell'attività Amazon ECS. Se stai eseguendo su Amazon EC2 (invece diAWS Fargate), potrebbe anche essere necessario modificare queste quote nell'istanza Amazon EC2 sottostante.

sysctl net

Un'altra categoria di impostazione sintonizzabile è la`sysctl`Impostazioni di rete. Dovresti fare riferimento alle impostazioni specifiche per la tua distribuzione Linux preferita. Molte di queste impostazioni regolano le dimensioni dei buffer di lettura e scrittura. Questo può aiutare in alcune situazioni quando si eseguono istanze Amazon EC2 su larga scala che hanno molti contenitori su di esse.

Best Practice - Scalabilità automatica e gestione della capacità

Amazon ECS viene utilizzato per eseguire carichi di lavoro di applicazioni containerizzate di tutte le dimensioni. Ciò include sia gli estremi degli ambienti di test minimi che gli ambienti di produzione di grandi dimensioni che operano su scala globale.

Con Amazon ECS, come tuttiAWSI prezzi dei servizi vengono calcolati in base all'uso effettivo. Se progettato in modo appropriato, è possibile risparmiare sui costi facendo in modo che l'applicazione utilizzi solo le risorse necessarie nel momento in cui ne ha bisogno. Questa guida alle best practice illustra come eseguire i carichi di lavoro Amazon ECS in modo che soddisfi i tuoi obiettivi di livello di servizio pur continuando a operare in modo conveniente.

Argomenti

Definizione delle dimensioni attività

Una delle scelte più importanti da fare quando si distribuiscono contenitori su Amazon ECS è la dimensione del contenitore e delle attività. Le dimensioni dei container e delle attività sono essenziali per la scalabilità e la pianificazione della capacità. In Amazon ECS, ci sono due metriche delle risorse utilizzate per la capacità: Memoria e CPU. La CPU è misurata in unità di 1/1024 di una vCPU completa (dove 1024 unità è uguale a 1 vCPU intera). La memoria viene misurata in megabyte. Nella definizione dell'attività è possibile dichiarare le prenotazioni e i limiti delle risorse.

Quando si dichiara una prenotazione, si dichiara la quantità minima di risorse richiesta da un'attività. L'attività riceve almeno la quantità di risorse richieste. L'applicazione potrebbe essere in grado di

utilizzare più CPU o memoria rispetto alla prenotazione dichiarata. Tuttavia, questo è soggetto a tutti i limiti che hai anche dichiarato. L'utilizzo di più dell'importo della prenotazione è noto come scoppio. In Amazon ECS, le prenotazioni sono garantite. Ad esempio, se utilizzi istanze Amazon EC2 per fornire capacità, Amazon ECS non inserisce un'attività in un'istanza in cui la prenotazione non può essere gestita.

Un limite è la quantità massima di unità CPU o memoria che il contenitore o l'attività può utilizzare. Qualsiasi tentativo di utilizzare più CPU di questo imit si traduce in limitazione. Qualsiasi tentativo di utilizzare più memoria comporta l'arresto del contenitore.

Scegliere questi valori può essere difficile. Questo perché i valori più adatti per l'applicazione dipendono in larga misura dai requisiti di risorse dell'applicazione. Il test di carico dell'applicazione è la chiave per una corretta pianificazione dei requisiti delle risorse e per una migliore comprensione dei requisiti dell'applicazione.

Applicazioni stateless

Per le applicazioni senza stato scalabili orizzontalmente, ad esempio un'applicazione dietro un bilanciamento del carico, è consigliabile innanzitutto determinare la quantità di memoria consumata dall'applicazione quando serve le richieste. A questo proposito, puoi utilizzare strumenti tradizionali come`psotopo` soluzioni di monitoraggio come CloudWatch Container Insights.

Quando si determina una prenotazione della CPU, considerare come si desidera ridimensionare l'applicazione in base ai requisiti aziendali. È possibile utilizzare prenotazioni CPU più piccole, ad esempio 256 unità CPU (o 1/4 vCPU), per ridimensionare in modo dettagliato e ridurre al minimo i costi. Tuttavia, potrebbero non scalare abbastanza velocemente da soddisfare picchi significativi della domanda. È possibile utilizzare prenotazioni CPU più grandi per scalare più rapidamente e quindi abbinare i picchi di domanda più rapidamente. Tuttavia, le prenotazioni CPU più grandi sono più costose.

Altre applicazioni

Per le applicazioni che non sono scalabili orizzontalmente, ad esempio lavoratori singleton o server di database, la capacità e i costi disponibili rappresentano le considerazioni più importanti. È consigliabile scegliere la quantità di memoria e CPU in base a quale test di carico indica che è necessario servire il traffico per raggiungere l'obiettivo del livello di servizio. Amazon ECS garantisce che l'applicazione sia posizionata su un host con capacità adeguata.

Configurazione del dimensionamento automatico del servizio

Un servizio Amazon ECS è una raccolta gestita di attività. Ogni servizio ha una definizione di attività associata, un conteggio delle attività desiderato e una strategia di posizionamento facoltativa. Il ridimensionamento automatico del servizio Amazon ECS viene implementato tramite il servizio di Application Auto Scaling. Application Auto Scaling utilizza le metriche CloudWatch come origine per il ridimensionamento delle metriche. Utilizza anche gli allarmi CloudWatch per impostare le soglie su quando scalare il servizio in entrata o in uscita. Fornire le soglie per la scalabilità, impostando una destinazione metrica, denominataMonitoraggio dei target, o specificando le soglie, denominateScaling di fasi: . Dopo aver configurato Application Auto Scaling, viene calcolato continuamente il conteggio delle attività desiderate appropriato per il servizio. Informa inoltre Amazon ECS quando il conteggio delle attività desiderato deve cambiare, ridimensionandolo o ridimensionandolo.

Per utilizzare in modo efficace la scalabilità automatica del servizio, è necessario scegliere una metrica di ridimensionamento appropriata. Discuteremo di come scegliere una metrica nelle sezioni seguenti.

Caratterizzazione dell'applicazione

Per ridimensionare correttamente un'applicazione è necessario conoscere le condizioni in cui l'applicazione deve essere ridimensionata e quando deve essere ridimensionata. In sostanza, un'applicazione dovrebbe essere ridimensionata se si prevede che la domanda superi la capacità. Al contrario, un'applicazione può essere ridimensionata per risparmiare i costi quando le risorse superano la domanda.

Identificazione di una metrica di utilizzo

Per scalare in modo efficace, è fondamentale identificare una metrica che indichi l'utilizzo o la saturazione. Questa metrica deve presentare le seguenti proprietà per essere utile per il ridimensionamento.

- La metrica deve essere correlata alla domanda. Quando le risorse vengono mantenute stabili, ma la domanda cambia, anche il valore della metrica deve cambiare. La metrica dovrebbe aumentare o diminuire quando la domanda aumenta o diminuisce.
- Il valore della metrica deve essere scalato in proporzione alla capacità. Quando la domanda rimane costante, l'aggiunta di altre risorse deve comportare una modifica proporzionale del valore della

metrica. Quindi, raddoppiando il numero di attività dovrebbe causare una diminuzione della metrica del 50%.

Il modo migliore per identificare una metrica di utilizzo consiste nel test del carico in un ambiente di pre-produzione, ad esempio un ambiente di gestione temporanea. Le soluzioni di test del carico commerciali e open-source sono ampiamente disponibili. Queste soluzioni in genere possono generare carichi sintetici o simulare il traffico utente reale.

Per avviare il processo di test del carico, è necessario iniziare creando dashboard per le metriche di utilizzo dell'applicazione. Queste metriche includono l'utilizzo della CPU, l'utilizzo della memoria, le operazioni I/O, la profondità della coda di I/O e la velocità effettiva di rete. Puoi raccogliere queste metriche con un servizio come CloudWatch Container Insights. In alternativa, utilizzare il servizio gestito Amazon per Prometheus insieme al servizio gestito Amazon per Grafana. Durante questo processo, assicurati di raccogliere e tracciare le metriche per i tempi di risposta o le percentuali di completamento del lavoro dell'applicazione.

Durante il test di carico, iniziare con una piccola richiesta o velocità di inserimento del lavoro. Mantenere questa velocità costante per alcuni minuti per consentire all'applicazione di riscaldarsi. Quindi, aumentare lentamente il tasso e tenerlo fermo per alcuni minuti. Ripetere questo ciclo, aumentando ogni volta la frequenza fino a quando i tempi di risposta o di completamento dell'applicazione non sono troppo lenti per soddisfare gli obiettivi dei livelli di servizio (SLOS).

Durante il test di carico, esaminare ciascuna delle metriche di utilizzo. Le metriche che aumentano insieme al carico sono i candidati migliori per servire come migliori metriche di utilizzo.

Quindi, identificare la risorsa che raggiunge la saturazione. Allo stesso tempo, esaminare anche le metriche di utilizzo per vedere quale si appiattisce per primo a un livello elevato. Oppure, esaminare quale raggiunge il picco e quindi arrestare prima l'applicazione in modo anomalo. Ad esempio, se l'utilizzo della CPU aumenta da 0% a 70 -80% mentre aggiungi carico, rimane a quel lebel dopo che viene aggiunto ancora più carico, allora è sicuro dire che la CPU è satura. A seconda dell'architettura della CPU, potrebbe non raggiungere mai il 100%. Ad esempio, si supponga che l'utilizzo della memoria aumenti man mano che si aggiunge il carico e quindi l'applicazione si blocca improvvisamente quando raggiunge l'attività o il limite di memoria dell'istanza di Amazon EC2. In questa situazione, è probabile che la memoria sia stata completamente consumata. Più risorse potrebbero essere utilizzate dall'applicazione. Di conseguenza, scegliere la metrica che rappresenta la risorsa che si esaurisce per prima.

Infine, prova di nuovo il test di caricamento dopo aver raddoppiato il numero di attività o istanze Amazon EC2. Si supponga che la metrica chiave aumenti, o diminuisce, a metà della velocità come prima. Se questo è il caso, la metrica è proporzionale alla capacità. Si tratta di una buona metrica di utilizzo per il ridimensionamento automatico.

Consideriamo ora questo scenario ipotetico. Si supponga di caricare testare un'applicazione e scoprire che l'utilizzo della CPU alla fine raggiunge l'80% a 100 richieste al secondo. Quando viene aggiunto più carico, non aumenta più l'utilizzo della CPU. Tuttavia, fa sì che l'applicazione risponda più lentamente. Quindi, si esegue di nuovo il test di carico, raddoppiando il numero di attività ma mantenendo la velocità al valore di picco precedente. Se si scopre che l'utilizzo medio della CPU scende a circa il 40%, l'utilizzo medio della CPU è un buon candidato per una metrica di ridimensionamento. D'altra parte, se l'utilizzo della CPU rimane all'80% dopo aver aumentato il numero di attività, l'utilizzo medio della CPU non è una buona metrica di ridimensionamento. In tal caso, è necessaria una maggiore ricerca per trovare una metrica adatta.

Modelli di applicazioni comuni e proprietà di ridimensionamento

Software di tutti i tipi sono eseguiti suAWS: . Molti carichi di lavoro sono fatti in casa, mentre altri sono basati su software open source più diffusi. Indipendentemente da dove provengono, abbiamo osservato alcuni modelli di progettazione comuni per i servizi. Come scalare efficacemente dipende in gran parte dal modello.

L'efficiente server associato alla CPU

L'efficiente server associato alla CPU non utilizza quasi risorse diverse dalla CPU e dal throughput di rete. Ogni richiesta può essere gestita dalla sola applicazione. Le richieste non dipendono da altri servizi, ad esempio i database. L'applicazione è in grado di gestire centinaia di migliaia di richieste simultanee e può utilizzare in modo efficiente più CPU per farlo. Ogni richiesta viene servita da un thread dedicato con un sovraccarico di memoria insufficiente oppure c'è un ciclo di eventi asincrono che viene eseguito su ogni CPU richiesta dal servizio. Ogni replica dell'applicazione è ugualmente in grado di gestire una richiesta. L'unica risorsa che potrebbe essere esaurita prima della CPU è la larghezza di banda della rete. Nei servizi limite della CPU, l'utilizzo della memoria, anche a velocità effettiva massima, è una frazione delle risorse disponibili.

Questo tipo di applicazione è adatto per il ridimensionamento automatico basato sulla CPU. L'applicazione gode della massima flessibilità in termini di scalabilità. Può essere scalato verticalmente fornendo istanze Amazon EC2 più grandi o vCPUs Fargate. Inoltre, può anche essere ridimensionato orizzontalmente aggiungendo più repliche. L'aggiunta di più repliche o il raddoppio delle dimensioni dell'istanza riduce la metà dell'utilizzo medio della CPU rispetto alla capacità.

Se utilizzi la capacità di Amazon EC2 per questa applicazione, considera la possibilità di inserirla in istanze ottimizzate per il calcolo, ad esempio`c5`o`c6g`Famiglia.

L'efficiente server associato alla memoria

L'efficiente server associato alla memoria alloca una quantità significativa di memoria per richiesta. Alla massima concorrenza, ma non necessariamente alla velocità effettiva, la memoria viene esaurita prima che le risorse della CPU vengano esaurite. La memoria associata a una richiesta viene liberata al termine della richiesta. Ulteriori richieste possono essere accettate fintanto che ci sia memoria disponibile.

Questo tipo di applicazione è adatto per il ridimensionamento automatico basato sulla memoria. L'applicazione gode della massima flessibilità in termini di scalabilità. Può essere scalato sia verticalmente fornendo risorse di memoria Amazon EC2 o Fargate più grandi ad esso. Inoltre, può anche essere ridimensionato orizzontalmente aggiungendo più repliche. L'aggiunta di più repliche o il raddoppio delle dimensioni dell'istanza può dimezzare l'utilizzo medio della memoria rispetto alla capacità.

Se utilizzi la capacità di Amazon EC2 per questa applicazione, considera la possibilità di inserirla in istanze ottimizzate per la memoria, ad esempio`r5`o`r6g`Famiglia.

Alcune applicazioni associate alla memoria non liberano la memoria associata a una richiesta al termine, in modo che una riduzione della concorrenza non comporti una riduzione della memoria utilizzata. A questo proposito, non è consigliabile utilizzare la scalabilità basata su memoria.

Il server basato sul lavoro

Il server basato sul lavoro elabora una richiesta per ogni singolo thread di lavoro una dopo l'altra. I thread di lavoro possono essere thread leggeri, come i thread POSIX. Possono anche essere thread di peso maggiore, come i processi UNIX. Indipendentemente dal thread che sono, c'è sempre una concorrenza massima che l'applicazione può supportare. Di solito il limite di concorrenza è impostato proporzionalmente alle risorse di memoria disponibili. Se viene raggiunto il limite di concorrenza, ulteriori richieste vengono inserite in una coda di backlog. Se la coda di backlog overflow, ulteriori richieste in entrata vengono immediatamente rifiutate. Le applicazioni più comuni che si adattano a questo modello includono il server web Apache e Gunicorn.

La concorrenza della richiesta è in genere la metrica migliore per ridimensionare questa applicazione. Poiché esiste un limite di concorrenza per ogni replica, è importante eseguire la scalabilità orizzontale prima di raggiungere il limite medio.

Il modo migliore per ottenere le metriche di concorrenza delle richieste consiste nel far sì che l'applicazione li riporti a CloudWatch. Ogni replica dell'applicazione può pubblicare il numero di richieste simultanee come metrica personalizzata ad alta frequenza. Si consiglia di impostare la frequenza almeno una volta al minuto. Dopo aver raccolto diversi report, è possibile utilizzare la concorrenza media come metrica di ridimensionamento. Questa metrica viene calcolata prendendo la concorrenza totale e dividendola per il numero di repliche. Ad esempio, se la concorrenza totale è 1000 e il numero di repliche è 10, la concorrenza media è 100.

Se l'applicazione si trova dietro un servizio di Application Load Balancer, è inoltre possibile utilizzare l'opzione`ActiveConnectionCount`per il bilanciamento del carico come fattore nella metrica di ridimensionamento. La`ActiveConnectionCount`deve essere divisa per il numero di repliche per ottenere un valore medio. Il valore medio deve essere utilizzato per la scalatura, anziché il valore del conteggio non elaborato.

Affinché questo progetto funzioni al meglio, la deviazione standard della latenza di risposta dovrebbe essere ridotta a basse velocità di richiesta. È consigliabile che, durante i periodi di bassa domanda, la maggior parte delle richieste ricevano risposta in breve tempo e non ci sono molte richieste che richiedono molto più tempo della media per rispondere. Il tempo medio di risposta dovrebbe essere vicino al tempo di risposta del 95° percentile. In caso contrario, potrebbe verificarsi un overflow della coda. Questo porta a errori. Si consiglia di fornire repliche aggiuntive, se necessario, per ridurre il rischio di overflow.

Il server di attesa

Il server in attesa esegue alcune elaborazioni per ogni richiesta, ma dipende fortemente da uno o più servizi downstream per funzionare. Le applicazioni contenitore spesso fanno uso intensivo di servizi downstream come database e altri servizi API. Questi servizi possono richiedere del tempo per rispondere, in particolare in scenari ad alta capacità o ad alta concorrenza. tCiò è dovuto al fatto che queste applicazioni tendono a utilizzare poche risorse CPU e la loro massima concorrenza in termini di memoria disponibile.

Il servizio di attesa è adatto sia nel modello server associato alla memoria che nel modello server basato sul lavoro, a seconda di come viene progettata l'applicazione. Se la concorrenza dell'applicazione è limitata solo dalla memoria, l'utilizzo medio della memoria deve essere utilizzato come metrica di ridimensionamento. Se la concorrenza dell'applicazione è basata su un limite di lavoro, la concorrenza media deve essere utilizzata come metrica di ridimensionamento.

Il server basato su Java

Se il server basato su Java è associato alla CPU e scalabile proporzionalmente alle risorse della CPU, potrebbe essere adatto per il modello di server associato alla CPU efficiente. In questo caso, l'utilizzo medio della CPU potrebbe essere appropriato come metrica di ridimensionamento. Tuttavia, molte applicazioni Java non sono legate alla CPU, il che le rende difficili da scalare.

Per prestazioni ottimali, ti consigliamo di allocare la maggior quantità di memoria possibile all'heap JVM (Java Virtual Machine). Le versioni recenti della JVM, incluso Java 8 update 191 o versioni successive, impostano automaticamente la dimensione dell'heap il più grande possibile per adattarla al contenitore. Ciò significa che, in Java, l'utilizzo della memoria è raramente proporzionale all'utilizzo delle applicazioni. Man mano che la velocità di richiesta e la concorrenza aumentano, l'utilizzo della memoria rimane costante. Per questo motivo, non è consigliabile scalare i server basati su Java in base all'utilizzo della memoria. In genere, si consiglia di ridimensionare l'utilizzo della CPU.

In alcuni casi, i server basati su Java incontrano esaurimento heap prima di esaurire la CPU. Se l'applicazione è soggetta a esaurimento dell'heap ad alta concorrenza, le connessioni medie sono la metrica di ridimensionamento migliore. Se l'applicazione è soggetta a esaurimento dell'heap a velocità effettiva elevata, la percentuale di richiesta media è la metrica di scalabilità migliore.

Server che utilizzano altri runtime garbage-collection

Molte applicazioni server sono basate su runtime che eseguono garbage collection come .NET e Ruby. Queste applicazioni server potrebbero inserirsi in uno dei modelli descritti in precedenza. Tuttavia, come per Java, non è consigliabile ridimensionare queste applicazioni in base alla memoria, poiché l'utilizzo medio della memoria osservato è spesso non correlato con la velocità effettiva o la concorrenza.

Per queste applicazioni, si consiglia di scalare l'utilizzo della CPU se l'applicazione è associata alla CPU. In caso contrario, si consiglia di scalare in base alla velocità effettiva media o alla concorrenza media, in base ai risultati dei test di carico.

Processor di Job

Molti carichi di lavoro comportano l'elaborazione asincrona dei processi. Includono applicazioni che non ricevono richieste in tempo reale, ma sottoscrivono invece una coda di lavoro per ricevere processi. Per questi tipi di applicazioni, la metrica di ridimensionamento corretta è quasi sempre la profondità della coda. La crescita della coda indica che il lavoro in sospeso supera la capacità di elaborazione, mentre una coda vuota indica che c'è più capacità del lavoro da svolgere.

AWSservizi di messaggistica, come Amazon SQS e Amazon Kinesis Data Streams, forniscono metriche CloudWatch che possono essere utilizzate per la scalabilità. Per Amazon SQS,`ApproximateNumberOfMessagesVisible`è la metrica migliore. Per i Kinesis Data Streams, prendere in considerazione l'utilizzo del`MillisBehindLatest`, pubblicata dalla Kinesis Client Library (KCL). Questa metrica deve essere calcolata sulla media di tutti i consumatori prima di utilizzarla per la scalabilità.

Capacità e disponibilità

La disponibilità delle applicazioni è fondamentale per offrire un'esperienza senza errori e per ridurre al minimo la latenza delle applicazioni. La disponibilità dipende dalla disponibilità di risorse accessibili e con capacità sufficiente per soddisfare la domanda.AWSfornisce diversi meccanismi per gestire la disponibilità. Per le applicazioni ospitate su Amazon ECS, queste includono la scalabilità automatica e le zone di disponibilità. La scalabilità automatica gestisce il numero di task o istanze in base alle metriche definite dall'utente, mentre le zone di disponibilità consentono di ospitare l'applicazione in posizioni isolate ma geograficamente chiuse.

Come per le dimensioni delle attività, la capacità e la disponibilità presentano alcuni compromessi da considerare. Idealmente, la capacità sarebbe perfettamente allineata con la domanda. Ci sarebbe sempre la capacità sufficiente per soddisfare le richieste e i processi di elaborazione per soddisfare gli obiettivi di livello di servizio (SLOS), tra cui una bassa latenza e un tasso di errore. La capacità non sarebbe mai troppo elevata, portando a costi eccessivi; né sarebbe mai troppo bassa, portando ad alti tassi di latenza e di errore.

La scalabilità automatica è un processo latente. Innanzitutto, le metriche in tempo reale devono essere consegnate a CloudWatch. Quindi, devono essere aggregati per l'analisi, che può richiedere fino a diversi minuti a seconda della granularità della metrica. CloudWatch confronta le metriche con le soglie di allarme per identificare una carenza o un eccesso di risorse. Per evitare l'instabilità, configurare gli allarmi in modo che la soglia impostata venga superata per alcuni minuti prima che l'allarme si spenga. Occorrono anche tempo per eseguire il provisioning di nuove attività e per terminare le attività che non sono più necessarie.

A causa di questi potenziali ritardi nel sistema descritto, è importante mantenere un po 'di spazio di crescita attraverso l'overprovisioning. Questa operazione può aiutare a soddisfare le esplosioni a breve termine della domanda. Questo aiuta anche l'applicazione a soddisfare richieste aggiuntive senza raggiungere la saturazione. Come buona pratica, è possibile impostare il target di ridimensionamento tra il 60 e l'80% dell'utilizzo. Ciò consente all'applicazione di gestire al meglio le esplosioni di domanda extra, mentre la capacità aggiuntiva è ancora in fase di provisioning.

Un altro motivo per cui si consiglia di eseguire il provisioning eccessivo è che è possibile rispondere rapidamente agli errori della zona di disponibilità.AWSconsiglia di gestire i carichi di lavoro di produzione da più zone di disponibilità. Questo perché, se si verifica un errore della zona di disponibilità, le attività in esecuzione nelle restanti zone di disponibilità possono comunque soddisfare la domanda. Se l'applicazione viene eseguita in due zone di disponibilità, è necessario raddoppiare il numero normale di attività. In questo modo è possibile fornire capacità immediata durante qualsiasi potenziale guasto. Se l'applicazione viene eseguita in tre zone di disponibilità, si consiglia di eseguire 1,5 volte il numero normale di attività. Cioè, esegui tre compiti per ogni due che sono necessari per servire ordinaria.

Massimizzazione della velocità di dimensionamento

La scalabilità automatica è un processo reattivo che richiede tempo per avere effetto. Tuttavia, ci sono alcuni modi per ridurre al minimo il tempo necessario per la scalabilità orizzontale.

Riduci le dimensioni dell'immagine Le immagini più grandi richiedono più tempo per scaricare da un repository di immagini e decomprimere. Pertanto, mantenere le dimensioni delle immagini più piccole riduce il tempo necessario per l'avvio di un contenitore. Per ridurre le dimensioni dell'immagine, è possibile seguire questi consigli specifici:

- Se puoi creare un binario statico o usare Golang, crea la tua immagine`FROM`scratch e includere solo l'applicazione binaria nell'immagine risultante.
- Usa immagini di base ridotte a icona provenienti da fornitori di distribuzione upstream, come Amazon Linux o Ubuntu.
- Non includere artefatti di compilazione nell'immagine finale. L'utilizzo di build multistadio può aiutare con questo.
- Compatto`RUN`stadi laddove possibile. Ogni`RUN`crea un nuovo livello immagine, portando ad un ulteriore round trip per scaricare il livello. Una singola`RUN`che ha più comandi uniti da`&&`ha meno livelli di uno con più`RUN`Stage.
- Se si desidera includere dati, ad esempio i dati di inferenza ML, nell'immagine finale, includere solo i dati necessari per avviare e iniziare a servire il traffico. Se recuperi i dati su richiesta da Amazon S3 o da altro spazio di archiviazione senza influire sul servizio, memorizza invece i dati in quei luoghi.

Tieni le tue immagini vicine. Maggiore è la latenza di rete, maggiore è il tempo necessario per scaricare l'immagine. Ospita le tue immagini in un repository nello stessoAWSArea in cui si trova il

carico di lavoro. Amazon ECR è un repository di immagini ad alte prestazioni disponibile in tutte le aree geografiche in cui Amazon ECS è disponibile. Evitare di attraversare Internet o un collegamento VPN per scaricare le immagini del contenitore. L'hosting delle immagini nella stessa regione migliora l'affidabilità complessiva. Riduce il rischio di problemi di connettività di rete e di disponibilità in un'area diversa. In alternativa, puoi anche implementare la replica tra aree Amazon ECR per aiutarti.

Riduzione delle soglie di controllo dello stato del sistema di bilanciamento del carico. I bilanciatori di carico eseguono controlli di integrità prima di inviare traffico all'applicazione. La configurazione predefinita del controllo di integrità per un gruppo target può richiedere 90 secondi o più. Durante questo, controlla lo stato di integrità e le richieste di ricezione. Abbassando l'intervallo di controllo dello stato e il conteggio delle soglie, l'applicazione accetta il traffico più rapidamente e riduce il carico su altre attività.

Considerate le prestazioni di avvio a freddo. Alcune applicazioni utilizzano runtime come Java eseguono la compilazione Just-In-Time (JIT). Il processo di compilazione almeno all'avvio può mostrare le prestazioni dell'applicazione. Una soluzione alternativa consiste nel riscrivere le parti critiche per la latenza del carico di lavoro in linguaggi che non impongono una penalizzazione delle prestazioni di avvio a freddo.

Utilizzare policy di dimensionamento di fase e non di monitoraggio dei target. Sono disponibili diverse opzioni di Application Auto Scaling delle applicazioni per le attività Amazon ECS. Il tracciamento target è la modalità più semplice da usare. Con esso, tutto ciò che devi fare è impostare un valore di destinazione per una metrica, ad esempio l'utilizzo medio della CPU. Quindi, il scaler automatico gestisce automaticamente il numero di attività necessarie per raggiungere tale valore. Tuttavia, si consiglia di utilizzare la scalabilità dei passaggi in modo da poter reagire più rapidamente ai cambiamenti della domanda. Con la scalabilità dei passaggi, è possibile definire le soglie specifiche per le metriche di ridimensionamento e il numero di attività da aggiungere o rimuovere quando le soglie vengono superate. E, cosa ancora più importante, è possibile reagire molto rapidamente ai cambiamenti della domanda riducendo al minimo il tempo in cui un allarme di soglia viene violato. Per ulteriori informazioni, consultaAuto Scaling del servizionellaAmazon Elastic Container Service: .

Se utilizzi istanze Amazon EC2 per fornire capacità cluster, considera i seguenti consigli:

Utilizza istanze Amazon EC2 più grandi e volumi Amazon EBS più veloci. Puoi migliorare la velocità di download e preparazione delle immagini utilizzando un'istanza Amazon EC2 più grande e un volume Amazon EBS più veloce. All'interno di una determinata famiglia di istanze Amazon EC2, la velocità effettiva massima di rete e Amazon EBS aumenta man mano che aumenta la dimensione dell'istanza (ad esempio, da`m5.xlarge`Da a`m5.2xlarge`). Inoltre, puoi anche personalizzare i volumi Amazon EBS per aumentarne la velocità effettiva e l'IOPS. Ad esempio, se usi`gp2`, utilizzare

volumi più grandi che offrono una maggiore velocità effettiva di base. Se usi`gp3`Specificare la velocità effettiva e IOPS al momento della creazione del volume.

Usa la modalità di rete bridge per le attività in esecuzione su istanze Amazon EC2. Attività che utilizzano`bridge`la modalità di rete su Amazon EC2 inizia più velocemente rispetto alle attività che utilizzano il`awsvpc`Modalità di rete. Quando`awsvpc`Amazon ECS collega un'elastic network interface (ENI) all'istanza prima di avviare l'attività. Questo introduce latenza aggiuntiva. Ci sono diversi compromessi per l'utilizzo della rete bridge però. Queste attività non hanno un proprio gruppo di sicurezza e ci sono alcune implicazioni per il bilanciamento del carico. Per ulteriori informazioni, consultaGruppi target del bilanciamento del cariconellaGuida utente per Elastic Load Balancing: .

Gestione degli shock della domanda

Alcune applicazioni subiscono improvvisi grandi shock della domanda. Questo accade per una serie di motivi: un evento di notizie, una grande vendita, un evento mediatico o qualche altro evento che diventa virale e provoca un rapido e significativo aumento del traffico in un lasso di tempo molto breve. Se non pianificato, la domanda può superare rapidamente le risorse disponibili.

Il modo migliore per gestire gli shock della domanda è anticiparli e pianificare di conseguenza. Poiché la scalabilità automatica può richiedere tempo, è consigliabile ridimensionare l'applicazione prima che inizi lo shock della domanda. Per ottenere risultati ottimali, si consiglia di disporre di un business plan che preveda una stretta collaborazione tra i team che utilizzano un calendario condiviso. Il team che sta pianificando l'evento dovrebbe lavorare a stretto contatto con il team responsabile della domanda in anticipo. Questo dà a quel team abbastanza tempo per avere un piano di pianificazione chiaro. Possono pianificare la capacità per la scalabilità orizzontale prima dell'evento e per la scalabilità successiva all'evento. Per ulteriori informazioni, consultaDimensionamento pianificatonellaApplication Auto Scaling User Guide: .

Se si dispone di un piano di Support Enterprise, assicurarsi di lavorare anche con il Technical Account Manager (TAM). Il TAM può verificare le quote di servizio e assicurarsi che tutte le quote necessarie vengano aumentate prima dell'inizio dell'evento. In questo modo, non colpisci accidentalmente le quote di servizio. Possono anche aiutarti preriscaldando servizi come i bilanciatori del carico per assicurarti che il tuo evento vada senza intoppi.

Gestire gli shock della domanda non programmati è un problema più difficile. Gli shock non programmati, se di ampiezza sufficientemente grande, possono causare rapidamente la domanda di superare la capacità. Può anche superare la capacità di reagire in scala automatica. Il modo migliore per prepararsi agli shock non programmati consiste nel fornire risorse eccessive. È necessario disporre di risorse sufficienti per gestire la massima richiesta di traffico prevista in qualsiasi momento.

Mantenere la massima capacità in previsione di shock della domanda non programmati può essere costoso. Per mitigare l'impatto sui costi, trovare una metrica o un evento indicatore leader che preveda un grande shock della domanda è imminente. Se la metrica o l'evento fornisce in modo affidabile un preavviso significativo, avviare immediatamente il processo di scalabilità orizzontale quando si verifica l'evento o quando la metrica supera la soglia specifica impostata.

Se l'applicazione è soggetta a improvvisi shock della domanda non pianificata, è consigliabile aggiungere una modalità ad alte prestazioni all'applicazione che sacrifichi funzionalità non critiche ma mantenga funzionalità cruciali per un cliente. Si supponga, ad esempio, che l'applicazione possa passare dalla generazione di costose risposte personalizzate alla pubblicazione di una pagina di risposta statica. In questo scenario, è possibile aumentare la velocità effettiva in modo significativo senza ridimensionare l'applicazione.

Infine, è possibile considerare la rottura dei servizi monolitici per gestire meglio gli shock della domanda. Se la tua applicazione è un servizio monolitico costoso da eseguire e lento a scalare, potresti essere in grado di estrarre o riscrivere pezzi critici per le prestazioni ed eseguirli come servizi separati. Questi nuovi servizi possono quindi essere scalati indipendentemente dai componenti meno critici. Avere la flessibilità necessaria per ridimensionare le funzionalità critiche per le prestazioni separatamente dalle altre parti dell'applicazione può ridurre il tempo necessario per aggiungere capacità e contribuire a ridurre i costi.

Capacità del cluster

In precedenza in questo argomento, abbiamo discusso come ridimensionare l'account di replica per l'account utilizzando le metriche di ridimensionamento. Le attività devono essere eseguite anche su risorse, incluse CPU e risorse di memoria. Questo riguarda ancora una volta il tema della capacità. In Amazon ECS, la capacità viene fornita tramite due fornitori principali:AWS Fargatee Amazon EC2.

Puoi fornire capacità a un cluster Amazon ECS in diversi modi. Ad esempio, puoi avviare istanze Amazon EC2 e registrarle con il cluster all'avvio utilizzando l'agente contenitore Amazon ECS. Tuttavia, questo metodo può essere impegnativo perché è necessario gestire autonomamente il ridimensionamento. Pertanto, ti consigliamo di utilizzare i provider di capacità Amazon ECS. Gestiscono la scalabilità delle risorse per te. Esistono tre tipi di provider di capacità: Amazon EC2, Fargate e Fargate Spot.

I fornitori di capacità Fargate e Fargate Spot gestiscono per voi il ciclo di vita delle attività di Fargate. Fargate fornisce capacità su richiesta e Fargate Spot fornisce capacità Spot. Quando viene avviata un'attività, ECS fornisce una risorsa Fargate per te. Questa risorsa Fargate viene fornita con le

unità di memoria e CPU che corrispondono direttamente ai limiti a livello di attività dichiarati nella definizione di attività. Ogni attività riceve la propria risorsa Fargate, creando una relazione 1:1 tra l'attività e il calcolo iresources.

Le attività eseguite su Fargate Spot sono soggette a interruzioni. Le interruzioni arrivano dopo un avvertimento di due minuti. Questi si verificano durante periodi di forte domanda. Fargate Spot funziona al meglio per carichi di lavoro a tolleranza di interruzione, come processi batch, ambienti di sviluppo o staging. Sono adatti anche per qualsiasi altro scenario in cui l'alta disponibilità e la bassa latenza non sono un requisito.

È possibile eseguire attività di Fargate Spot insieme alle attività su richiesta di Fargate. Usandoli insieme, ricevi la capacità di provisioning «burst» a un costo inferiore.

ECS può anche gestire la capacità di istanza Amazon EC2 per le tue attività. Ogni fornitore di capacità Amazon EC2 è associato a un gruppo di Auto Scaling Amazon EC2 specificato. Quando utilizzi Amazon EC2 Capacity Provider, il Auto Scaling del cluster ECS mantiene le dimensioni del gruppo di ridimensionamento automatico Amazon EC2 per garantire che tutte le attività pianificate possano essere eseguite.

Best practice relative alla capacità del cluster

Aggiungi spazio al tuo servizio, non al provider di capacità. I fornitori di capacità Amazon EC2 offrono un valore di capacità target. Se imposti il valore inferiore al 100%, ECS predispone più istanze Amazon EC2 del necessario per gestire le tue attività. Avere diverse istanze di Amazon EC2 pronte per accettare le attività può essere utile. Tuttavia, quando usi Amazon Virtual Private Cloud, l'avvio di nuove attività richiede tempo aggiuntivo per scaricare l'immagine e collegare un'interfaccia di rete. Questa latenza aggiunta potrebbe essere dannosa per la tua linea di fondo.

Pertanto, ti consigliamo di procedere come segue. Anziché ridurre la capacità di destinazione del provider di capacità, aumentare il numero di repliche nel servizio modificando la metrica di ridimensionamento del tracciamento di destinazione o le soglie di ridimensionamento dei passaggi del servizio. Per ulteriori informazioni sui criteri di dimensionamento correlati, consultaPolicy di dimensionamento di monitoraggio obiettivioPolicy di dimensionamento per fasinellaAmazon Elastic Container Service: . Amazon EC2 Capacity Provider fornisce la capacità necessaria per attività aggiuntive aggiungendo istanze aggiuntive al gruppo di Auto Scaling. Ciò consente di garantire che sia le risorse di elaborazione che le risorse delle applicazioni siano disponibili quando ne hai bisogno. Ad esempio, può aiutare raddoppiando il numero di attività in un servizio ECS per soddisfare un aumento immediato del 100% della domanda.

Scegliere le dimensioni delle attività Fargate

Se esegui le tue attività suAWS FargateÈ necessario dichiarare i limiti di CPU e memoria di attività nella definizione di attività. ECS utilizza questi limiti per determinare il tipo di istanza di Fargate su cui eseguire l'attività. I limiti stabiliti devono essere maggiori o uguali a eventuali prenotazioni dichiarate. Nella maggior parte dei casi, è possibile impostarli sulla somma delle prenotazioni di ogni contenitore dichiarato nella definizione dell'attività. Quindi, anche arrotondare il numero fino alla dimensione dell'istanza di Fargate più vicina. Per ulteriori informazioni sulle dimensioni disponibili, consultaMemoria e CPU di attivitànellaAmazon Elastic Container Service: .

Scelta del tipo di istanza Amazon EC2

Se utilizzi Amazon EC2 per fornire capacità per il tuo cluster ECS, puoi scegliere tra un'ampia selezione di tipi di istanza. Tutti i tipi e le famiglie di istanze Amazon EC2 sono compatibili con ECS.

Per determinare quali tipi di istanza è possibile utilizzare, eliminare i tipi di istanza o le famiglie di istanze che non soddisfano i requisiti specifici dell'applicazione. Ad esempio, se l'applicazione richiede una GPU, è possibile escludere qualsiasi tipo di istanza che non dispone di una GPU. Tuttavia, dovresti anche prendere in considerazione altri requisiti. Ad esempio, considerare l'architettura della CPU, la velocità effettiva di rete e se l'archiviazione delle istanze è un requisito. Esaminare quindi la quantità di CPU e memoria fornita da ciascun tipo di istanza. Come regola generale, la CPU e la memoria devono essere sufficientemente grandi da contenere almeno una replica dell'attività che si desidera eseguire.

È possibile scegliere tra i tipi di istanza compatibili con l'applicazione. Con istanze di dimensioni maggiori, puoi avviare più attività contemporaneamente. Inoltre, con istanze più piccole, è possibile scalare in modo più dettagliato per risparmiare sui costi. Non è necessario scegliere un singolo tipo di istanza Amazon EC2 che si adatti a tutte le applicazioni del cluster. È invece possibile creare più gruppi Auto Scaling. Ogni gruppo può avere un tipo di istanza diverso. Quindi, puoi creare un fornitore di capacità Amazon EC2 per ciascuno di questi gruppi. Infine, nella strategia Provider di capacità del tuo servizio e attività, puoi selezionare il Provider di capacità che meglio si adatta alle sue esigenze.

Utilizzo di Amazon EC2 Spot e FARGATE_SPOT

La capacità spot può offrire risparmi significativi sui costi rispetto alle istanze on-demand. La capacità spot è un eccesso di capacità che ha un prezzo significativamente inferiore rispetto alla capacità

su richiesta o riservata. La capacità spot è adatta per i carichi di lavoro di elaborazione batch e apprendimento automatico, nonché per gli ambienti di sviluppo e gestione temporanea. Più in generale, è adatto a qualsiasi carico di lavoro che tollera tempi di inattività temporanei.

Comprendere che le seguenti conseguenze perché la capacità spot potrebbe non essere sempre disponibile.

- In primo luogo, durante periodi di domanda estremamente elevata, la capacità spot potrebbe non essere disponibile. Ciò può causare il ritardo dell'attività di Fargate Spot e dei lanci di istanza di Amazon EC2 Spot. In questi eventi, i servizi ECS tentano di avviare nuovamente le attività e i gruppi di Auto Scaling Amazon EC2 tentano di avviare nuovamente le istanze, finché non diventa disponibile la capacità richiesta. Fargate e Amazon EC2 non sostituiscono la capacità Spot con la capacità on demand.
- In secondo luogo, quando aumenta la domanda complessiva di capacità, le istanze e le attività Spot potrebbero essere terminate con un avviso di soli due minuti. Dopo l'invio dell'avviso, le attività dovrebbero iniziare un arresto ordinato, se necessario, prima che l'istanza venga terminata completamente. Ciò aiuta a ridurre al minimo la possibilità di errori. Per ulteriori informazioni su uno spegnimento regolare, consultaArresti aggraziati con ECS: .

Per ridurre al minimo le carenze di capacità spot, prendere in considerazione i seguenti suggerimenti:

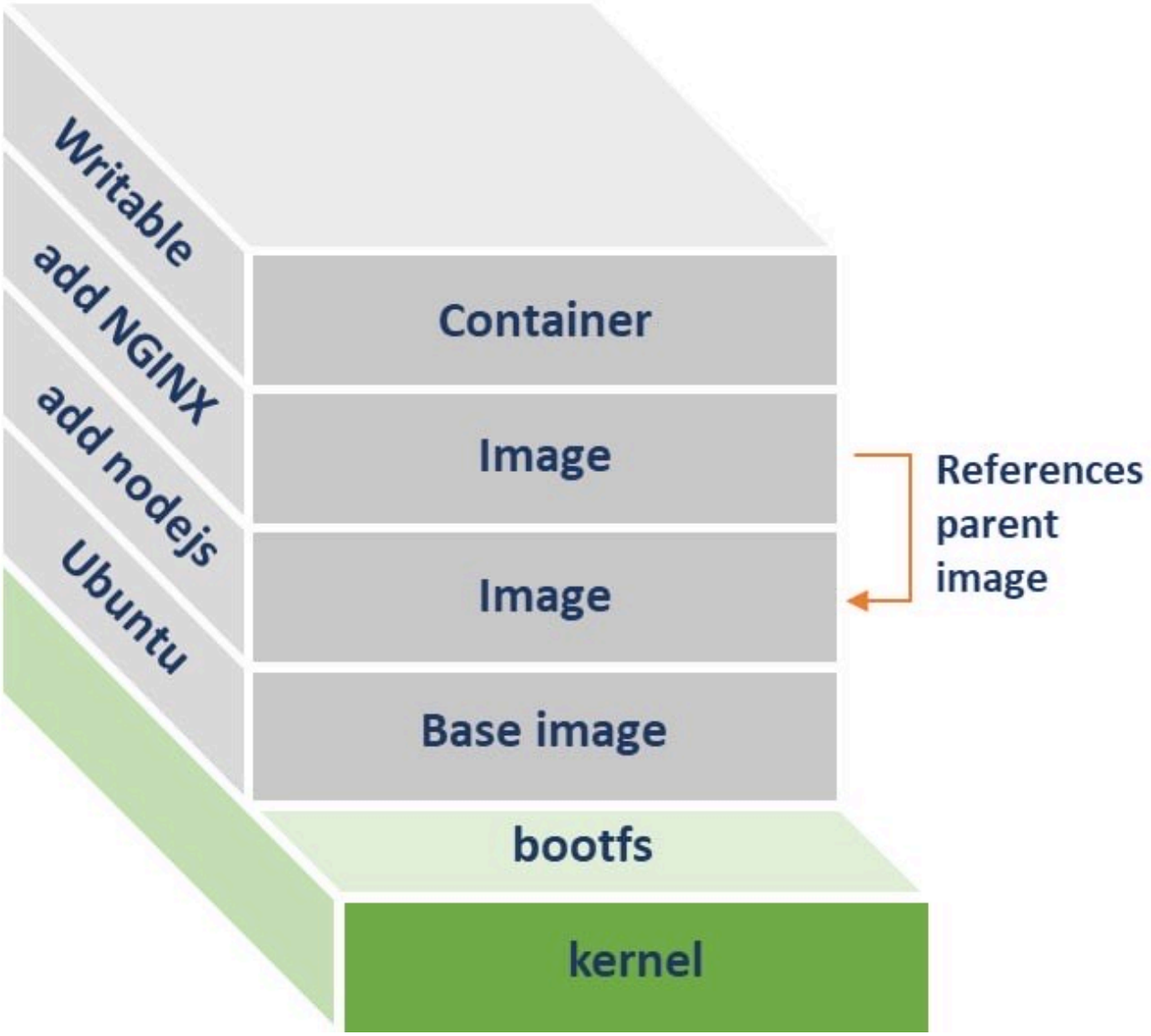
- Utilizzare più regioni e zone di disponibilità. La capacità spot varia in base alla regione e alla zona di disponibilità. È possibile migliorare la disponibilità spot eseguendo i carichi di lavoro in più aree e zone di disponibilità. Se possibile, specificare le subnet in tutte le zone di disponibilità nelle regioni in cui si eseguono le attività e le istanze.
- Utilizza più tipi di istanza Amazon EC2. Quando utilizzi criteri di istanza misti con Amazon EC2 Auto Scaling, vengono lanciati più tipi di istanza nel tuo gruppo di ridimensionamento automatico. Ciò garantisce che una richiesta di capacità Spot possa essere soddisfatta quando necessario. Per massimizzare l'affidabilità e ridurre al minimo la complessità, utilizzare i tipi di istanza con circa la stessa quantità di CPU e memoria nei criteri Istanze miste. Queste istanze possono provenire da una generazione diversa o varianti dello stesso tipo di istanza di base. Si noti che potrebbero essere fornite con funzionalità aggiuntive che potrebbero non essere necessarie. Un esempio di tale elenco potrebbe includere m4.large, m5.large, m5a.large, m5d.large, m5n.large, m5dn.large e m5ad.large. Per ulteriori informazioni, consultare la sezione relativa ai Gruppi Auto Scaling con più tipi di istanze e opzioni di acquisto nella Guida per l'utente di Amazon EC2 Auto Scaling.
- Utilizzare la strategia di allocazione spot ottimizzata per la capacità. Con Amazon EC2 Spot, puoi scegliere tra le strategie di allocazione ottimizzate per la capacità e i costi. Se scegli la strategia

ottimizzata per la capacità quando avvii una nuova istanza, Amazon EC2 Spot seleziona il tipo di istanza con la massima disponibilità nella Zona di disponibilità selezionata. Ciò consente di ridurre la possibilità che l'istanza venga terminata subito dopo l'avvio.

Procedure ottimali - Archiviazione persistente

Puoi utilizzare Amazon ECS per eseguire applicazioni containerizzate con stato su larga scala utilizzandoAWSservizi di archiviazione, come Amazon EFS, Amazon EBS o Amazon FSx for Windows File Server, che forniscono la persistenza dei dati a contenitori intrinsecamente effimeri. Il terminePersistenza dei datisignifica che i dati stessi durano il processo che li ha creati. Persistenza dei dati inAWSsi ottiene accoppiando i servizi di elaborazione e storage. Analogamente a Amazon EC2, puoi anche utilizzare Amazon ECS per disaccoppiare il ciclo di vita delle applicazioni containerizzate dai dati che consumano e producono. Utilizzo diAWS, le attività Amazon ECS possono mantenere i dati anche dopo la fine delle attività.

Per impostazione predefinita, i contenitori non mantengono i dati che producono. Quando un contenitore viene terminato, i dati che ha scritto al suo livello scrivibile vengono distrutti con il contenitore. Ciò rende i contenitori adatti per applicazioni stateless che non hanno bisogno di archiviare dati localmente. Le applicazioni containerizzate che richiedono la persistenza dei dati necessitano di un back-end di archiviazione che non viene distrutto quando il contenitore dell'applicazione termina.



Un'immagine contenitore è costruita su una serie di livelli. Ogni livello rappresenta un'istruzione nel Dockerfile da cui è stata creata l'immagine. Ogni layer è di sola lettura, ad eccezione del contenitore. Cioè, quando si crea un contenitore, un livello scrivibile viene aggiunto sui livelli sottostanti. Tutti i file creati dal contenitore, eliminati o modificati vengono scritti nel layer scrivibile. Quando il contenitore termina, anche il livello scrivibile viene eliminato simultaneamente. Un nuovo contenitore che utilizza la stessa immagine ha un proprio livello scrivibile. Questo layer non include alcuna modifica. Pertanto, i dati di un contenitore devono sempre essere memorizzati al di fuori del livello scrivibile contenitore.

Con Amazon ECS, puoi eseguire contenitori con stato utilizzando volumi. Amazon ECS è integrato nativamente con Amazon EFS e utilizza volumi integrati con Amazon EBS. Per i contenitori Windows,

Amazon ECS si integra con Amazon FSx for Windows File Server per fornire spazio di archiviazione persistente.

Argomenti

Scegliere il giusto tipo di storage per i container

Le applicazioni in esecuzione in un cluster Amazon ECS possono utilizzare una varietà diAWSServizi di storage e prodotti di terze parti per fornire storage persistente per carichi di lavoro con stato. È consigliabile scegliere il back-end di storage per l'applicazione containerizzata in base all'architettura e ai requisiti di storage dell'applicazione. Per ulteriori informazioni suAWSservizi di archiviazione, vedereArchiviazione nel cloudAWS: .

Per i cluster Amazon ECS che contengono istanze Linux o sono utilizzati con Fargate, Amazon ECS si integra con Amazon EFS e Amazon EBS per fornire lo storage dei contenitori. La differenza più distintiva tra Amazon EFS e Amazon EBS è che è possibile montare simultaneamente un filesystem Amazon EFS su migliaia di attività Amazon ECS. Al contrario, i volumi Amazon EBS non supportano l'accesso simultaneo. In considerazione di ciò, Amazon EFS è l'opzione di archiviazione consigliata per le applicazioni containerizzate scalabili orizzontalmente. Questo perché supporta la concorrenza. Amazon EFS memorizza i dati in modo ridondante in più zone di disponibilità e offre un accesso a bassa latenza dalle attività Amazon ECS, indipendentemente dalla zona di disponibilità. Amazon EFS supporta attività eseguite sia su Amazon EC2 che su Fargate.

Supponiamo di avere un'applicazione come un database transazionale che richiede una latenza inferiore al millisecondo e non abbia bisogno di un filesystem condiviso quando è scalato orizzontalmente. Per tale applicazione, ti consigliamo di utilizzare i volumi Amazon EBS per lo storage persistente. Attualmente, Amazon ECS supporta i volumi Amazon EBS solo per le attività ospitate su Amazon EC2. Il Support per i volumi Amazon EBS non è disponibile per le attività su Fargate. Prima di utilizzare i volumi Amazon EBS con le attività Amazon ECS, devi prima allegare i volumi Amazon EBS alle istanze del contenitore e gestire i volumi separatamente dal ciclo di vita dell'attività.

Per i cluster che contengono istanze di Windows, Amazon FSx for Windows File Server fornisce spazio di archiviazione permanente per i contenitori. Amazon FSx for Windows File Server supporta distribuzioni multi-AZ. Attraverso queste distribuzioni, è possibile condividere un filesystem con attività Amazon ECS in esecuzione su più zone di disponibilità.

Puoi anche utilizzare l'archiviazione istanze Amazon EC2 per la persistenza dei dati per le attività Amazon ECS ospitate su Amazon EC2 utilizzando supporti bind o volumi Docker. Quando si utilizzano supporti bind o volumi Docker, i contenitori memorizzano i dati nel file system dell'istanza contenitore. Una limitazione dell'utilizzo di un filesystem host per lo storage dei contenitori è che i dati sono disponibili solo su una singola istanza contenitore alla volta. Ciò significa che i contenitori possono essere eseguiti solo sull'host in cui risiedono i dati. Pertanto, l'utilizzo dello storage host è consigliato solo in scenari in cui la replica dei dati viene gestita a livello di applicazione.

Volumi Amazon EFS

Amazon Elastic File System (Amazon EFS) offre un file system NFS elastico, semplice, scalabile e completamente gestito. È progettato per essere in grado di scalare on demand fino a petabyte senza interrompere le applicazioni. Si può ridimensionare in o out mentre si aggiungono e rimuovono i file.

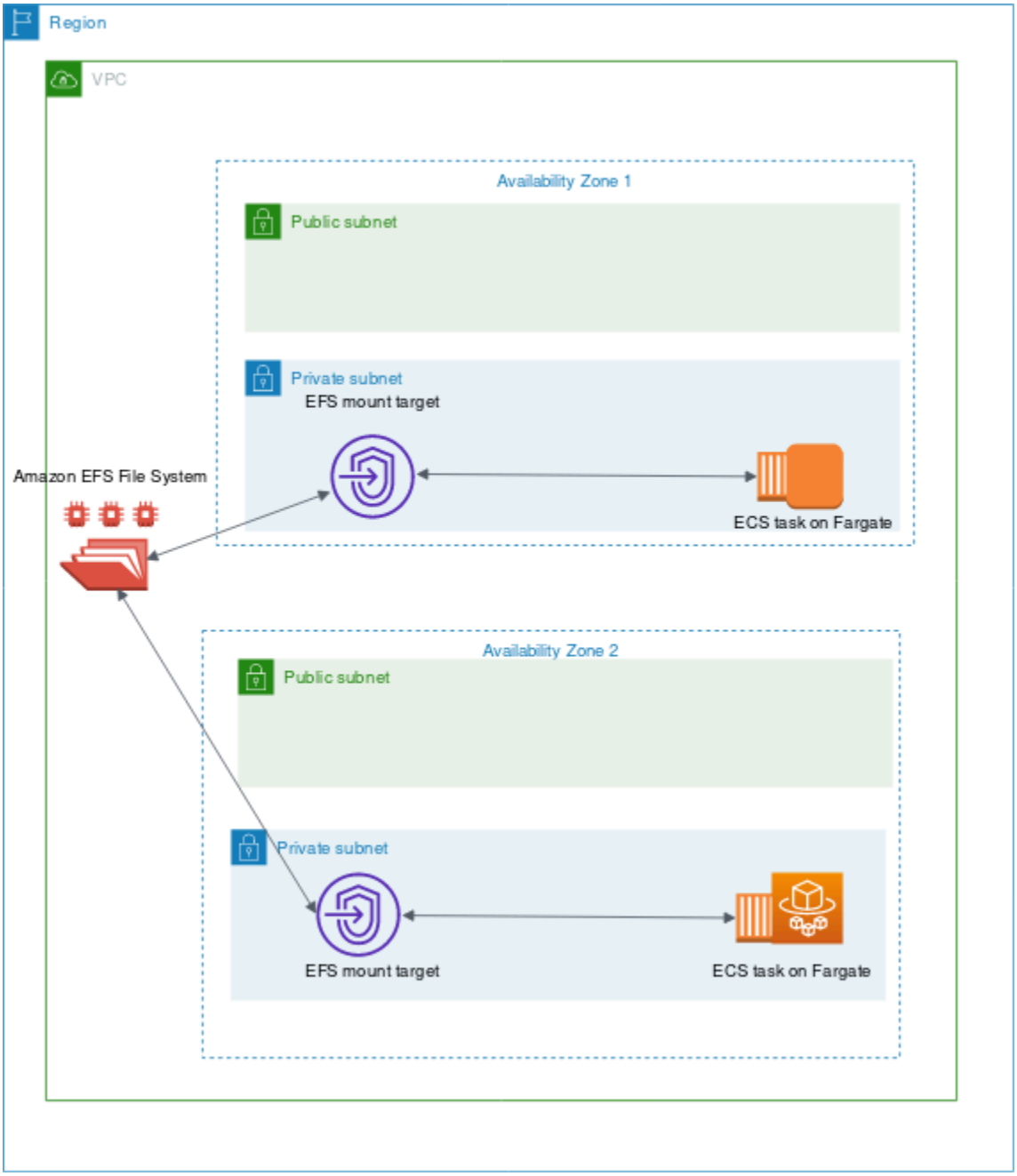
Puoi eseguire le tue applicazioni con stato in Amazon ECS utilizzando volumi Amazon EFS per fornire spazio di archiviazione persistente. Attività Amazon ECS eseguite su istanze Amazon EC2 o su Fargate utilizzando la versione della piattaforma`1.4.0`e versioni successive possono montare un file system Amazon EFS esistente. Dato che più contenitori possono montare e accedere simultaneamente a un file system EFS Amazon, le tue attività hanno accesso allo stesso set di dati indipendentemente da dove sono ospitate.

Per montare un file system Amazon EFS nel container, puoi fare riferimento al file system Amazon EFS e al punto di montaggio del container nella definizione dell'attività di Amazon ECS. Di seguito è riportato un frammento di una definizione di attività che utilizza Amazon EFS per l'archiviazione di container.

```
...
"containerDefinitions":[
    {
        "mountPoints": [
            {
                "containerPath": "/opt/my-app",
                 "sourceVolume": "Shared-EFS-Volume"
            }
```

```
    }
  ]
...
"volumes": [
    {
      "efsVolumeConfiguration": {
        "fileSystemId": "fs-1234",
        "transitEncryption": "DISABLED",
        "rootDirectory": ""
      },
      "name": "Shared-EFS-Volume"
    }
  ]
```

Amazon EFS memorizza i dati in modo ridondante in più zone di disponibilità all'interno di una singola regione. Un'attività Amazon ECS monta il file system Amazon EFS utilizzando un target di montaggio Amazon EFS nella sua zona di disponibilità. Un'attività Amazon ECS può montare un file system Amazon EFS solo se il filesystem Amazon EFS ha un target di mount nella zona di disponibilità in cui viene eseguita l'attività. Pertanto, si consiglia di creare obiettivi di montaggio EFS Amazon in tutte le zone di disponibilità in cui si prevede di ospitare attività Amazon ECS.



Per ulteriori informazioni, consultaVolumi Amazon EFSnellaGuida per sviluppatori Amazon Elastic Container Service: .

Controllo di sicurezza e accesso

Amazon EFS offre funzioni di controllo degli accessi che puoi utilizzare per garantire che i dati archiviati in un file system EFS di Amazon siano sicuri e accessibili solo dalle applicazioni che ne hanno bisogno. Puoi proteggere i dati abilitando la crittografia dei dati inattivi e in transito. Per ulteriori

informazioni, consultaCrittografia dati in Amazon EFSnellaGuida per l'utente Amazon Elastic File System: .

Oltre alla crittografia dei dati, puoi anche utilizzare Amazon EFS per limitare l'accesso a un file system. Esistono tre modi per implementare il controllo di accesso in EFS.

- Gruppi di sicurezza: con le destinazioni di montaggio EFS di Amazon, puoi configurare un gruppo di sicurezza utilizzato per consentire e negare il traffico di rete. È possibile configurare il gruppo di sicurezza collegato ad Amazon EFS per consentire il traffico NFS (porta 2049) dal gruppo di sicurezza collegato alle istanze Amazon ECS o, quando si utilizza il`awsvpc`modalità di rete, l'attività Amazon ECS.
- IAM: puoi limitare l'accesso a un file system Amazon EFS mediante IAM. Una volta configurate, le attività Amazon ECS richiedono un ruolo IAM per l'accesso al file system per montare un file system EFS. Per ulteriori informazioni, consultaUtilizzo di IAM per controllare l'accesso ai dati del file systemnellaGuida per l'utente Amazon Elastic File System: .

  I criteri IAM possono inoltre applicare condizioni predefinite, ad esempio richiedere a un client di utilizzare TLS durante la connessione a un file system Amazon EFS. Per ulteriori informazioni, consultaChiavi di condizione EFS Amazon per i clientinellaGuida per l'utente Amazon Elastic File System: .
- Punti di accesso Amazon EFS: I punti di accesso Amazon EFS sono punti di accesso specifici dell'applicazione in un file system Amazon EFS. È possibile utilizzare i punti di accesso per applicare un'identità utente, inclusi i gruppi POSIX dell'utente, per tutte le richieste al file system effettuate tramite il punto di accesso. I punti di accesso possono inoltre applicare una directory radice diversa per il file system. In questo modo i client possono accedere solo ai dati nella directory specificata o nelle sue sottodirectory.

Considera l'implementazione di tutti e tre i controlli di accesso su un file system EFS Amazon per la massima sicurezza. Ad esempio, puoi configurare il gruppo di sicurezza associato a un mount point Amazon EFS in modo da consentire solo l'ingresso del traffico NFS da un gruppo di sicurezza associato all'istanza del contenitore o all'attività Amazon ECS. Inoltre, è possibile configurare Amazon EFS in modo che richieda un ruolo IAM per accedere al file system, anche se la connessione proviene da un gruppo di sicurezza consentito. Infine, è possibile utilizzare i punti di accesso EFS di Amazon per applicare le autorizzazioni utente POSIX e specificare le directory radice per le applicazioni.

Il seguente frammento di definizione dell'attività mostra come montare un file system Amazon EFS utilizzando un punto di accesso.

```
"volumes": [
    {
      "efsVolumeConfiguration": {
        "fileSystemId": "fs-1234",
        "authorizationConfig": {
          "acessPointId": "fsap-1234",
          "iam": "ENABLED"
        },
        "transitEncryption": "ENABLED",
        "rootDirectory": ""
      },
      "name": "my-filesystem"
    }
]
```

Performance

Amazon EFS offre due modalità di prestazioni: Generic Purpose e max I/O. General Purpose è adatto per applicazioni sensibili alla latenza come sistemi di gestione dei contenuti e strumenti CI/CD. Al contrario, i file system Max I/O sono adatti per carichi di lavoro quali analisi dei dati, elaborazione dei supporti e apprendimento automatico. Questi carichi di lavoro devono eseguire operazioni parallele da centinaia o addirittura migliaia di container e richiedono il massimo throughput aggregato possibile e IOPS. Per ulteriori informazioni, consultaModalità di performance di Amazon EFSnellaGuida per l'utente Amazon Elastic File System: .

Alcuni carichi di lavoro sensibili alla latenza richiedono sia i livelli I/O più elevati forniti dalla modalità prestazionale Max I/O sia la latenza inferiore fornita dalla modalità prestazionale per uso generico. Per questo tipo di carico di lavoro, consigliamo di creare più file system in modalità prestazionale per uso generico. In questo modo, puoi distribuire il carico di lavoro dell'applicazione su tutti questi file system, purché il carico di lavoro e le applicazioni possano supportarlo.

Throughput

A tutti i file system EFS di Amazon è associato un throughput misurato determinato dalla quantità di throughput di cui è stato eseguito il provisioning per i file system utilizzandoThroughput assegnatoo la quantità di dati memorizzati nella classe di archiviazione EFS Standard o One Zone per i file system

che utilizzanoThroughput di rottura: . Per ulteriori informazioni, consultaInformazioni sulla velocità effettiva misuratanellaGuida per l'utente Amazon Elastic File System: .

La modalità velocità effettiva predefinita per i file system EFS di Amazon è la modalità di frammentazione. Con la modalità di frammentazione, la velocità effettiva disponibile per un file system viene scalata in entrata o in uscita man mano che un file system cresce. Poiché i carichi di lavoro basati su file in genere aumentano e richiedono livelli di throughput elevati per periodi di tempo e livelli di throughput inferiori per il resto del tempo, Amazon EFS è progettato per consentire livelli di throughput elevati per periodi di tempo. Inoltre, poiché molti carichi di lavoro sono pesanti in lettura, le operazioni di lettura vengono misurate con un rapporto di 1:3 rispetto ad altre operazioni NFS (come la scrittura).

Tutti i file system EFS di Amazon offrono prestazioni di base costanti di 50 MB/s per ogni TB di storage Amazon EFS Standard o Amazon EFS One Zone. Tutti i file system (indipendentemente dalle dimensioni) possono arrivare a 100 MB/s. I file system con più di 1 TB di storage EFS Standard o EFS One Zone possono arrivare a 100 MB/s per ogni TB. Poiché le operazioni di lettura sono misurate con un rapporto di 1:3, è possibile guidare fino a 300 MIB/s per ogni TIB di velocità effettiva di lettura. Man mano che aggiungi dati al file system, la velocità effettiva massima disponibile per il file system viene scalata in modo lineare e automatico con lo spazio di archiviazione nella classe di archiviazione Amazon EFS Standard. Se è necessario un throughput superiore a quello che è possibile ottenere con la quantità di dati archiviati, è possibile configurare Throughput con provisioning in base alla quantità specifica richiesta dal carico di lavoro.

La velocità effettiva del file system viene condivisa tra tutte le istanze Amazon EC2 connesse a un file system. Ad esempio, un file system da 1 TB che può passare a 100 MB/s di velocità effettiva può guidare 100 MB/s da una singola istanza di Amazon EC2 può ogni unità 10 MB/s. Per ulteriori informazioni, consultaPersistEFS di AmazonnellaGuida per l'utente Amazon Elastic File System: .

Ottimizzazione dei costi

Amazon EFS semplifica la scalabilità dello storage per te. I file system Amazon EFS crescono automaticamente man mano che aggiungi più dati. Specialmente con Amazon EFSThroughput di rottura, la velocità effettiva su Amazon EFS ridimensiona in base alle dimensioni del file system nella classe di storage standard. Per migliorare la velocità effettiva senza pagare un costo aggiuntivo per la velocità effettiva di provisioning su un filesystem EFS, puoi condividere un file system EFS di Amazon con più applicazioni. Utilizzando i punti di accesso EFS Amazon, è possibile implementare l'isolamento dello storage nei file system Amazon EFS condivisi. In questo modo, anche se le

applicazioni condividono ancora lo stesso file system, non possono accedere ai dati a meno che non li autorizzi.

Man mano che i dati crescono, Amazon EFS ti aiuta a spostare automaticamente i file a cui si accede raramente in una classe di archiviazione inferiore. La classe di storage di Amazon EFS Standard-Infrequent Access (IA) riduce i costi di storage per i file ai quali non viene effettuato l'accesso ogni giorno. Ciò avviene senza sacrificare disponibilità elevata, durabilità elevata, elasticità e accesso al file system POSIX forniti da Amazon EFS. Per ulteriori informazioni, consultaClassi di storage Amazon EFSnellaGuida per l'utente Amazon Elastic File System: .

Valuta la possibilità di utilizzare le politiche del ciclo di vita di Amazon EFS per risparmiare automaticamente, spostando i file a cui si accede raramente nello spazio di archiviazione EFS IA. Per ulteriori informazioni, consultaGestione del ciclo di vita di Amazon EFSnellaGuida per l'utente Amazon Elastic File System: .

Quando crei un file system Amazon EFS, puoi scegliere se Amazon EFS replica i tuoi dati in più zone di disponibilità (Standard) o memorizza i dati in modo ridondante all'interno di un'unica zona di disponibilità. La classe di archiviazione Amazon EFS One Zone può ridurre i costi di stoccaggio di un margine significativo rispetto alle classi di archiviazione Amazon EFS Standard. Considera l'utilizzo della classe di archiviazione Amazon EFS One Zone per carichi di lavoro che non richiedono resilienza multi-AZ. È possibile ridurre ulteriormente il costo dello spazio di archiviazione di Amazon EFS One Zone spostando i file a cui si accede raramente in Amazon EFS One Zone-Infrequent Access. Per ulteriori informazioni, consultaAccesso non frequente ad Amazon EFS: .

Protezione dei dati

Amazon EFS memorizza i dati in modo ridondante in più zone di disponibilità per i file system che utilizzano classi di archiviazione Standard. Se selezioni classi di archiviazione Amazon EFS One Zone, i dati vengono archiviati in modo ridondante all'interno di un'unica zona di disponibilità. Inoltre, Amazon EFS è progettato per fornire una durata pari al 99,9999999% (11 9) in un determinato anno.

Come per qualsiasi ambiente, è consigliabile disporre di un backup e creare misure di protezione contro l'eliminazione accidentale. Per i dati EFS di Amazon, tale best practice include un backup funzionante e regolarmente testato utilizzandoAWS Backup: . I file system che utilizzano le classi di archiviazione Amazon EFS One Zone sono configurati per eseguire automaticamente il backup dei file durante la creazione del file system, a meno che non si scelga di disabilitare questa funzionalità. Per ulteriori informazioni, consultaProtezione dei dati per Amazon EFSnellaGuida per l'utente Amazon Elastic File System: .

Casi d'uso

Amazon EFS fornisce accesso condiviso parallelo che aumenta e diminuisce automaticamente in base ai file che vengono aggiunti o rimossi. Di conseguenza, Amazon EFS è adatto a qualsiasi applicazione che richiede uno spazio di archiviazione con funzionalità come bassa latenza, velocità effettiva elevata e coerenza di lettura dopo scrittura. Amazon EFS è un back-end di archiviazione ideale per applicazioni che scalano orizzontalmente e richiedono un file system condiviso. I carichi di lavoro, ad esempio l'analisi dei dati, l'elaborazione multimediale, la gestione di contenuti e la distribuzione di contenuti via Web, sono alcuni dei casi d'uso comuni di Amazon EFS.

Un caso d'uso in cui Amazon EFS potrebbe non essere adatto è per le applicazioni che richiedono una latenza inferiore al millisecondo. Questo è generalmente un requisito per i sistemi di database transazionali. Ti consigliamo di eseguire test delle prestazioni dello storage per determinare l'impatto dell'utilizzo di Amazon EFS per le applicazioni sensibili alla latenza. Se le prestazioni delle applicazioni si riducono quando si utilizza Amazon EFS, prendere in considerazione Amazon EBS io2 Block Express, che fornisce latenza I/O a bassa varianza inferiore al millisecondo sulle istanze Nitro. Per ulteriori informazioni, consulta Tipi di volume Amazon EBS nella Guida per l'utente di Amazon EC2 per le istanze Linux.

Alcune applicazioni non riescono se lo storage sottostante viene modificato in modo imprevisto. Pertanto, Amazon EFS non è la scelta migliore per queste applicazioni. Piuttosto, è preferibile utilizzare un sistema di storage che non consente l'accesso simultaneo da più postazioni.

Volumi Docker

I volumi Docker sono una funzionalità del runtime del contenitore Docker che consente ai contenitori di mantenere i dati montando una directory dal filesystem dell'host. I driver del volume Docker (detti anche plugin) vengono utilizzati per integrare i volumi di container con sistemi di storage esterni, ad esempio Amazon EBS. I volumi Docker sono supportati solo se si ospitano attività di Amazon ECS su istanze di Amazon EC2.

Le attività Amazon ECS possono utilizzare i volumi Docker per mantenere i dati utilizzando i volumi Amazon EBS. Questo viene fatto allegando un volume Amazon EBS a un'istanza Amazon EC2 e quindi montando il volume in un'attività utilizzando volumi Docker. Un volume Docker può essere condiviso tra più attività Amazon ECS sull'host.

La limitazione dei volumi Docker è che il file system utilizzato dall'attività è legato all'istanza specifica di Amazon EC2. Se l'istanza si interrompe per qualsiasi motivo e l'attività viene posizionata su

un'altra istanza, i dati vengono persi. È possibile assegnare task alle istanze per garantire che i volumi EBS associati siano sempre disponibili per le attività.

Per ulteriori informazioni, consultaVolumi DockernellaGuida per sviluppatori Amazon Elastic Container Service: .

Ciclo di vita dei volumi Amazon EBS

Ci sono due modelli di utilizzo chiave con lo stoccaggio del contenitore e Amazon EBS. Il primo è quando un'applicazione deve mantenere i dati e prevenire la perdita di dati quando il suo contenitore termina. Un esempio di questo tipo di applicazione sarebbe un database transazionale come MySQL. Quando un'attività MySQL termina, è previsto che un'altra attività lo sostituisca. In questo scenario, il ciclo di vita del volume è separato dal ciclo di vita dell'attività. Quando si utilizza EBS per mantenere i dati del contenitore, è consigliabile utilizzare vincoli di posizionamento delle attività per limitare il posizionamento dell'attività a un singolo host con il volume EBS collegato.

Il secondo è quando il ciclo di vita del volume è indipendente dal ciclo di vita dell'attività. Ciò è particolarmente utile per le applicazioni che richiedono storage ad alte prestazioni e bassa latenza, ma che non richiedono la persistenza dei dati al termine dell'attività. Ad esempio, un carico di lavoro ETL che elabora grandi volumi di dati può richiedere uno storage a throughput elevato. Amazon EBS è adatto a questo tipo di carico di lavoro in quanto fornisce volumi ad alte prestazioni che forniscono fino a 256.000 IOPS. Quando l'attività termina, la replica sostitutiva può essere posizionata in modo sicuro su qualsiasi host Amazon EC2 del cluster. Finché l'attività ha accesso a un back-end di archiviazione in grado di soddisfare i requisiti di prestazioni, l'attività può svolgere la propria funzione. Pertanto, in questo caso non sono necessari vincoli di posizionamento delle attività.

Se alle istanze Amazon EC2 del cluster sono associati più tipi di volumi Amazon EBS, puoi utilizzare i vincoli di posizionamento delle attività per assicurarti che le attività vengano collocate in istanze con un volume Amazon EBS appropriato associato. Si supponga, ad esempio, che un cluster abbia alcune istanze con un`gp2`, mentre altri usano`io1`volumi. È possibile allegare attributi personalizzati alle istanze con`io1`e quindi utilizzare i vincoli di posizionamento delle attività per garantire che le attività a uso intensivo di I/O siano sempre posizionate su istanze contenitore con`io1`volumi.

I seguentiAWS CLIcomando viene utilizzato per posizionare gli attributi su un'istanza di container di Amazon ECS.

```
aws ecs put-attributes \
     --attributes name=EBS,value=io1,targetId=<your-container-instance-arn>
```

Disponibilità dei dati Amazon EBS

I contenitori sono in genere di breve durata, creati di frequente e terminati con la scalabilità orizzontale delle applicazioni. Come procedura consigliata, è possibile eseguire carichi di lavoro in più zone di disponibilità per migliorare la disponibilità delle applicazioni. Amazon ECS ti offre un modo per controllare il posizionamento delle attività utilizzando strategie di posizionamento delle attività e vincoli di posizionamento delle attività. Quando un carico di lavoro persiste i suoi dati utilizzando volumi Amazon EBS, le sue attività devono essere collocate nella stessa zona di disponibilità del volume Amazon EBS. Si consiglia inoltre di impostare un vincolo di posizionamento che limiti l'area di disponibilità in cui è possibile inserire un'attività. In questo modo, le attività e i volumi corrispondenti si trovino sempre nella stessa zona di disponibilità.

Quando si eseguono attività autonome, è possibile controllare quale zona di disponibilità viene inserita impostando vincoli di posizionamento utilizzando l'attributo zona di disponibilità.

```
attribute:ecs.availability-zone == us-east-1a
```

Quando si eseguono applicazioni che potrebbero trarre vantaggio dall'esecuzione in più zone di disponibilità, è consigliabile creare un servizio Amazon ECS diverso per ciascuna zona di disponibilità. In questo modo, le attività che richiedono un volume Amazon EBS vengano sempre posizionate nella stessa zona di disponibilità del volume associato.

Ti consigliamo di creare istanze di container in ogni zona di disponibilità, allegando volumi EBS Amazon utilizzandoModelli di lancioe aggiungendoAttributi personalizzatialle istanze per differenziarle dalle altre istanze del contenitore nel cluster Amazon ECS. Durante la creazione di servizi, configura i vincoli di posizionamento delle attività per assicurarti che Amazon ECS inserisca le attività nella zona di disponibilità e nell'istanza corrette. Per ulteriori informazioni, consultaEsempi di vincoli di posizionamento attivitànellaGuida per sviluppatori Amazon Elastic Container Service: .

Plug-in volume Docker

I plugin Docker come Portworx forniscono un'astrazione tra il volume Docker e il volume Amazon EBS. Questi plugin possono creare dinamicamente un volume Amazon EBS quando inizia l'attività che richiede un volume. Portworx può anche collegare un volume a un nuovo host quando un contenitore termina e la sua replica successiva viene posizionata su un'istanza contenitore diversa. Replica inoltre i dati del volume di ciascun contenitore tra i nodi Amazon ECS e tra le zone di disponibilità. Per ulteriori informazioni, consultaPortworx: .

Amazon FSx for Windows File Server

Amazon FSx for Windows File Server offre storage di file completamente gestito, altamente affidabile e scalabile accessibile tramite il protocollo SMB (Server Message Block) standard del settore. È basato su Windows Server e offre un'ampia gamma di funzionalità amministrative, quali le quote utente, il ripristino dei file dell'utente finale e l'integrazione con Microsoft Active Directory (AD). Offre opzioni di implementazione singola e multiAZ, backup completamente gestiti e crittografia dei dati inattivi e in transito.

Amazon ECS supporta l'utilizzo di Amazon FSx for Windows File Server nelle definizioni delle attività di Amazon ECS Windows che consentono l'archiviazione persistente come punto di montaggio tramite protocollo SMBv3 utilizzando una funzione SMB chiamata GlobalMapping.

Per configurare l'integrazione Amazon FSx for Windows File Server e Amazon ECS, l'istanza del contenitore di Windows deve essere un membro di dominio in un servizio di dominio Active Directory, ospitato da unAWS Directory Service for Microsoft Active Directory, Active Directory locale o Active Directory self-hosted su Amazon EC2.AWS Secrets Managerviene utilizzato per archiviare dati sensibili come il nome utente e la password di una credenziale di Active Directory che viene utilizzata per mappare la condivisione nell'istanza del contenitore di Windows.

Per utilizzare Amazon FSx for Windows File Server per i volumi di file system per i container, devi specificare le configurazioni del volume e del punto di montaggio nella definizione dell'attività. Di seguito è riportato un frammento di una definizione di attività che utilizza Amazon FSx for Windows File Server per l'archiviazione di container.

```
{
 "containerDefinitions": [{
  "name": "container-using-fsx",
  "image": "iis:2",
  "entryPoint": [
   "powershell",
   "-command"
  ],
  "mountPoints": [{
   "sourceVolume": "myFsxVolume",
   "containerPath": "\\mount\\fsx",
   "readOnly": false
  }]
 }],
 "volumes": [{
```

```
  "fsxWindowsFileServerVolumeConfiguration": {
   "fileSystemId": "fs-ID",
   "authorizationConfig": {
    "domain": "ADDOMAIN.local",
    "credentialsParameter": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:SecretName"
   },
   "rootDirectory": "share"
  }
 }]
}
```

Per ulteriori informazioni, consultaVolumi Amazon FSx for Windows File ServernellaGuida per sviluppatori Amazon Elastic Container Service: .

Controllo di sicurezza e accesso

Amazon FSx for Windows File Server offre le seguenti funzioni di controllo dell'accesso che puoi utilizzare per garantire che i dati memorizzati in un file system Amazon FSx for Windows File Server siano protetti e accessibili solo dalle applicazioni che ne necessitano.

Crittografia dei dati

Amazon FSx for Windows File Server supporta due forme di crittografia per i file system. Sono la crittografia dei dati in transito e la crittografia dei dati memorizzati su disco. La crittografia dei dati in transito è supportata nelle condivisioni file mappate su un'istanza contenitore che supporta il protocollo SMB 3.0 o versioni successive. La crittografia dei dati inattivi viene abilitata automaticamente quando si crea un file system Amazon FSx. Amazon FSx crittografa automaticamente i dati in transito utilizzando la crittografia SMB durante l'accesso al file system senza la necessità di modificare le applicazioni. Per ulteriori informazioni, consultaCrittografia dei dati in Amazon FSxnellaGuida per l'utente di Amazon FSx for Windows File Server: .

Controllo di accesso a livello di cartella tramite ACL di Windows

L'istanza di Windows Amazon EC2 accede alle condivisioni file Amazon FSx utilizzando le credenziali di Active Directory. Utilizza elenchi di controllo di accesso (ACL) standard di Windows per il controllo di accesso a grana fine a livello di file e cartella. È possibile creare più credenziali, ognuna per una cartella specifica all'interno della condivisione che esegue il mapping a un'attività specifica.

Nell'esempio seguente, l'attività ha accesso alla cartellaApp01utilizzando una credenziale salvata in Secrets Manager. Il suo Amazon Resource Name (ARN) è1234: .

```
"rootDirectory": "\\path\\to\\my\\data\App01",
"credentialsParameter": "arn-1234",
"domain": "corp.fullyqualified.com",
```

In un altro esempio, un'attività ha accesso alla cartellaApp02utilizzando una credenziale salvata in Secrets Manager. Il suo ARN è 6789.

```
"rootDirectory": "\\path\\to\\my\\data\App02",
"credentialsParameter": "arn-6789",
"domain": "corp.fullyqualified.com",
```

Casi d'uso

I contenitori non sono progettati per mantenere i dati. Tuttavia, alcune applicazioni .NET containerizzate potrebbero richiedere cartelle locali come archiviazione persistente per salvare gli output dell'applicazione. Amazon FSx for Windows File Server offre una cartella locale nel container. Ciò consente a più contenitori di leggere e scrivere sullo stesso file system supportato da una condivisione SMB.

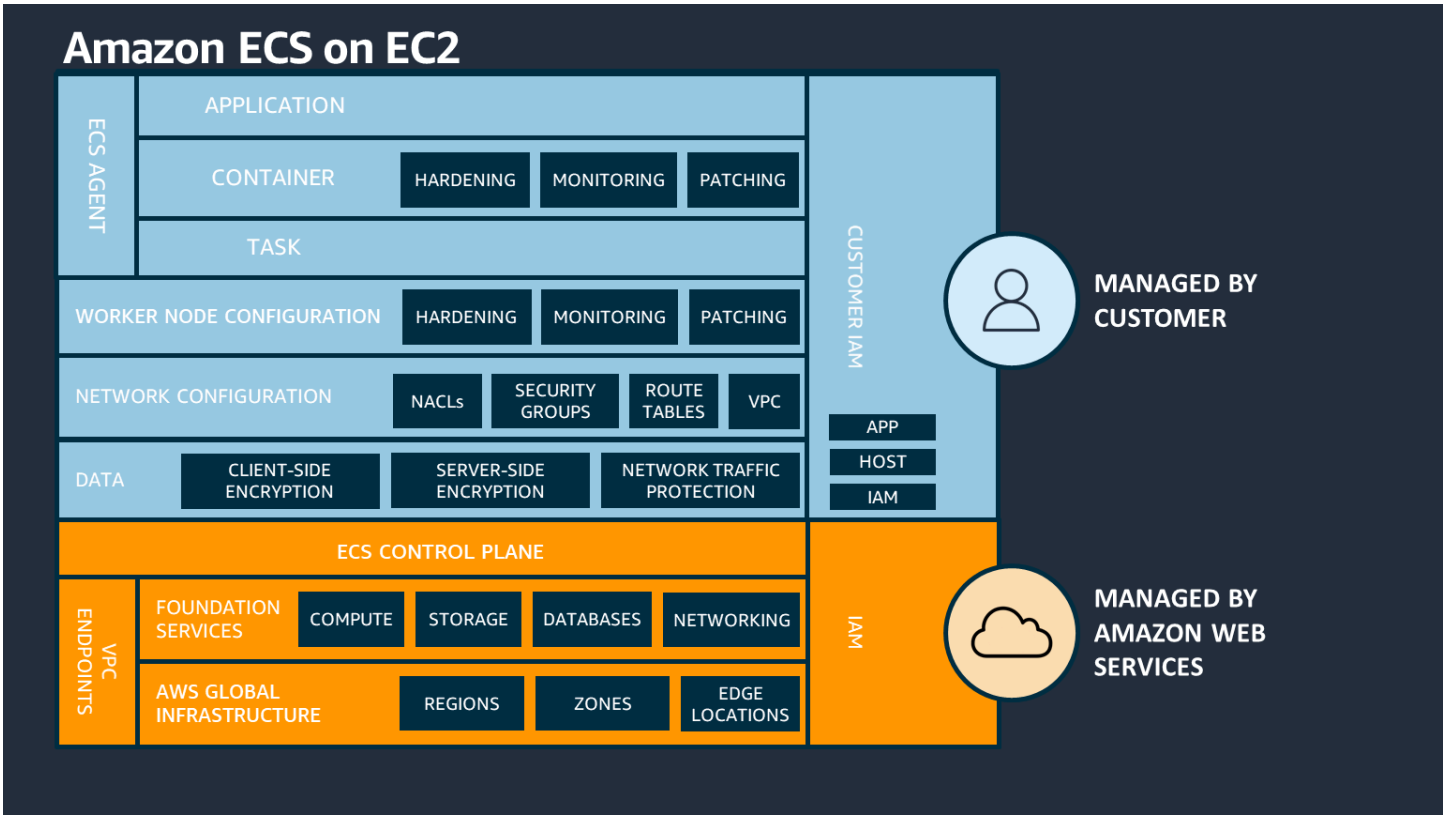
Best practice - Sicurezza

Questa guida fornisce consigli sulla sicurezza e sulla conformità per proteggere le informazioni, i sistemi e le altre risorse che si basano su Amazon ECS. Vengono inoltre introdotte alcune valutazioni dei rischi e strategie di mitigazione che è possibile utilizzare per avere una migliore presa sui controlli di sicurezza creati per i cluster Amazon ECS e i carichi di lavoro supportati. Ogni argomento di questa guida inizia con una breve panoramica, seguita da un elenco di consigli e best practice che puoi utilizzare per proteggere i tuoi cluster Amazon ECS.

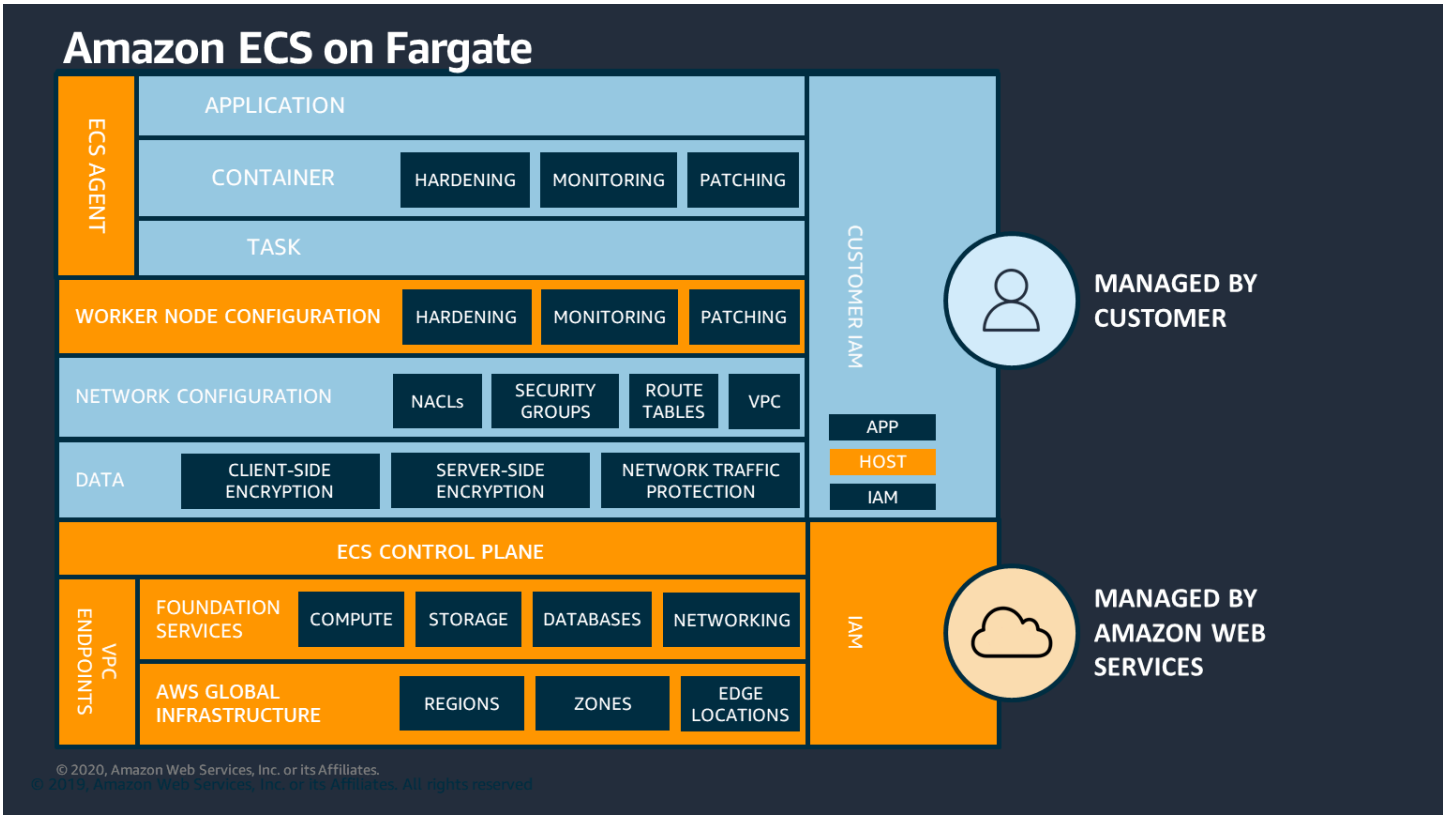
Argomenti

Modello di responsabilità condivisa

La sicurezza e la conformità di un servizio gestito come Amazon ECS sono una responsabilità condivisa sia per te che perAWS: . In generale,AWSè responsabile della sicurezza «del» cloud, mentre tu, il cliente, sei responsabile della sicurezza «nel» cloud.AWSè responsabile della gestione del piano di controllo Amazon ECS, inclusa l'infrastruttura necessaria per fornire un servizio sicuro e affidabile. Inoltre, sei in gran parte responsabile degli argomenti contenuti in questa guida. Ciò include la sicurezza dei dati, della rete e del runtime, nonché la registrazione e il monitoraggio.



Per quanto riguarda la sicurezza delle infrastrutture,AWSassume più responsabilità perAWS Fargatedi quanto non faccia per altre istanze autogestite. Con Fargate,AWSgestisce la sicurezza dell'istanza sottostante nel cloud e il runtime utilizzato per eseguire le attività. Fargate ridimensiona automaticamente la vostra infrastruttura per conto vostro.



Prima di estendere i servizi al cloud, devi capire quali aspetti della sicurezza e della conformità sei responsabile.

Per ulteriori informazioni sul modello di responsabilità condivisa, consulta.Modello di responsabilità condivisa: .

AWS Identity and Access Management

È possibile utilizzareAWS Identity and Access Management(IAM) per gestire e controllare l'accesso al tuoAWSservizi e risorse tramite criteri basati su regole per scopi di autenticazione e autorizzazione. In particolare, attraverso questo servizio, puoi controllare l'accesso alAWSutilizzando policy applicate a utenti, gruppi o ruoli IAM. Tra questi tre, gli utenti IAM sono account che possono avere accesso alle risorse. E, un ruolo IAM è un insieme di autorizzazioni che possono essere assunte da un'identità autenticata, che non è associata a una particolare identità esterna a IAM. Per ulteriori informazioni, consultaCriteri e autorizzazioni in IAM?: .

Gestione dell'accesso ad Amazon ECS

Puoi controllare l'accesso ad Amazon ECS creando e applicando le politiche IAM. Questi criteri sono composti da un insieme di azioni che si applicano a un set specifico di risorse. L'azione di un criterio

definisce l'elenco delle operazioni (ad esempio le API Amazon ECS) consentite o negate, mentre la risorsa controlla quali sono gli oggetti Amazon ECS a cui si applica l'azione. È possibile aggiungere condizioni a un criterio per restringere l'ambito di applicazione. Ad esempio, è possibile scrivere un criterio per consentire l'esecuzione di un'azione solo su attività con un determinato set di tag. Per ulteriori informazioni, consultaFunzionamento di Amazon ECS con IAMnellaGuida per sviluppatori di Amazon Elastic Container: .

Recommendations

Ti consigliamo anche di completare le seguenti operazioni durante la configurazione dei ruoli e dei policy IAM.

Seguire la politica di accesso meno privilegiato

Creare criteri con ambito per consentire agli utenti di eseguire i processi prescritti. Ad esempio, se uno sviluppatore deve interrompere periodicamente un'attività, creare un criterio che consenta solo quella particolare azione. L'esempio seguente consente solo a un utente di interrompere un'attività che appartiene a un particolare`task_family`In un cluster con un Amazon Resource Name (ARN) specifico. Fare riferimento a un ARN in una condizione è anche un esempio di utilizzo di autorizzazioni a livello di risorsa. Puoi utilizzare le autorizzazioni a livello di risorsa per specificare la risorsa a cui si desidera applicare un'operazione.

> **Note**
>
> Quando si fa riferimento a un ARN in un criterio, utilizzare il nuovo formato ARN più lungo. Per ulteriori informazioni, consultaAmazon Resource Name (ARN) e IDnellaGuida per sviluppatori di Amazon Elastic Container: .

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ecs:StopTask"
      ],
      "Condition": {
        "ArnEquals": {
```

```
          "ecs:cluster": "arn:aws:ecs:<region>:<aws_account_id>:cluster/<cluster_name>"
        }
      },
      "Resource": [
        "arn:aws:ecs:<region>:<aws_account_id>:task-definition/<task_family>:*"
      ]
    }
  ]
}
```

Lasciare che la risorsa cluster funga da limite amministrativo

I criteri con ambito troppo ristretto possono causare una proliferazione di ruoli e aumentare il sovraccarico amministrativo. Anziché creare ruoli con ambito solo per attività o servizi specifici, creare ruoli con ambito per cluster e utilizzare il cluster come limite amministrativo principale.

Isolare gli utenti finali dall'API Amazon ECS creando pipeline automatizzate

Puoi limitare le azioni che gli utenti possono utilizzare creando pipeline che impacchettano e distribuiscono automaticamente le applicazioni nei cluster Amazon ECS. Questo delega efficacemente il processo di creazione, aggiornamento ed eliminazione delle attività alla pipeline. Per ulteriori informazioni, consultaTutorial: Distribuzione standard Amazon ECS con CodePipelinenellaAWS CodePipelineGuida per l'utente: .

Utilizzare le condizioni dei criteri per un ulteriore livello di sicurezza

Quando è necessario un ulteriore livello di sicurezza, aggiungere una condizione al criterio. Ciò può essere utile se si sta eseguendo un'operazione con privilegi o quando è necessario limitare l'insieme di azioni che possono essere eseguite su determinate risorse. Il criterio di esempio seguente richiede l'autorizzazione a più fattori quando si elimina un cluster.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ecs:DeleteCluster"
      ],
      "Condition": {
        "Bool": {
```

```
          "aws:MultiFactorAuthPresent": "true"
        }
      },
    "Resource": ["*"]
    }
  ]
}
```

I tag applicati ai servizi vengono propagate a tutte le attività che fanno parte del servizio. Per questo motivo, puoi creare ruoli che rientrano nell'ambito delle risorse Amazon ECS con tag specifici. Nel criterio seguente, un'entità IAM avvia e arresta tutte le attività con una chiave tag`Departmente` un tag-value di`Accounting`: .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ecs:StartTask",
                "ecs:StopTask",
                "ecs:RunTask"
            ],
            "Resource": "arn:aws:ecs:*",
            "Condition": {
                "StringEquals": {"ecs:ResourceTag/Department": "Accounting"}
            }
        }
    ]
}
```

Controlla periodicamente l'accesso alle API Amazon ECS

Un utente potrebbe modificare i ruoli. Dopo aver modificato i ruoli, le autorizzazioni concesse in precedenza potrebbero non essere più applicabili. Assicurati di controllare chi ha accesso alle API Amazon ECS e se tale accesso è ancora giustificato. Considerare l'integrazione di IAM con una soluzione di gestione del ciclo di vita degli utenti che revoca automaticamente l'accesso quando un utente lascia l'organizzazione. Per ulteriori informazioni, consultaLinee guida sugli audit di sicurezza Amazon ECSnellaInformazioni generali su Amazon Web Services: .

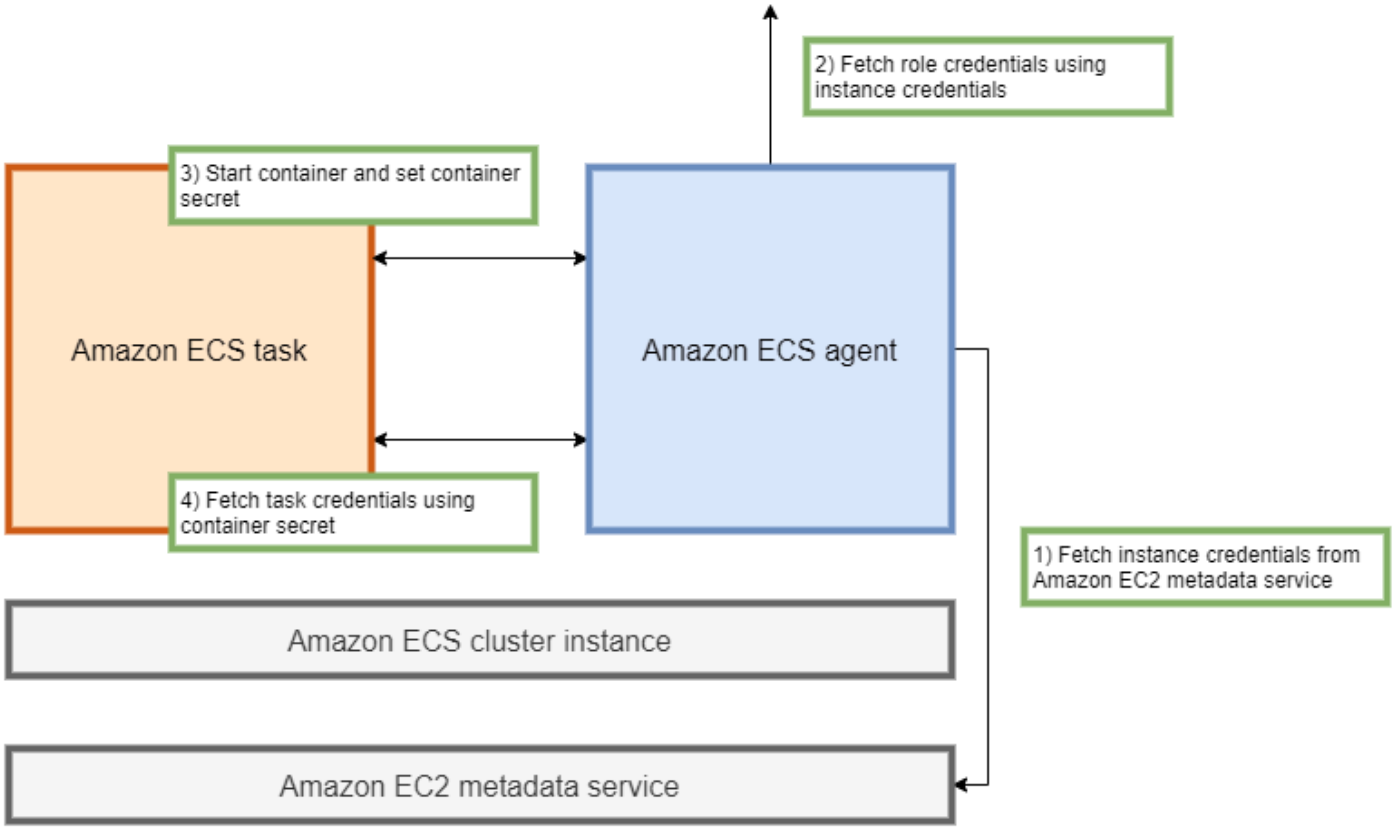
Utilizzo dei ruoli IAM con le attività Amazon ECS

Si consiglia di assegnare a un'attività un ruolo IAM. Il suo ruolo può essere distinto dal ruolo dell'istanza Amazon EC2 su cui è in esecuzione. Assegnando a ogni attività un ruolo si allinea con il principio dell'accesso meno privilegiato e consente un maggiore controllo granulare su azioni e risorse.

Quando si assegnano ruoli IAM per un'attività, è necessario utilizzare il criterio di attendibilità seguente in modo che ciascuna delle attività possa assumere un ruolo IAM diverso da quello utilizzato dall'istanza EC2. In questo modo, l'attività non eredita il ruolo dell'istanza EC2.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "ecs-tasks.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Quando aggiungi un ruolo di attività a una definizione di attività, l'agente contenitore Amazon ECS crea automaticamente un token con un ID credenziali univoco (ad esempio`12345678-90ab-cdef-1234-567890abcdef`) per l'attività. Questo token e le credenziali del ruolo vengono quindi aggiunti alla cache interna dell'agente. L'agente popola la variabile di ambiente`AWS_CONTAINER_CREDENTIALS_RELATIVE_URI`nel contenitore con l'URI dell'ID credenziale (ad esempio`/v2/credentials/12345678-90ab-cdef-1234-567890abcdef`).



È possibile recuperare manualmente le credenziali del ruolo temporaneo dall'interno di un contenitore aggiungendo la variabile di ambiente all'indirizzo IP dell'agente contenitore Amazon ECS ed eseguendo il comando`curl`sulla stringa risultante.

```
curl 192.0.2.0$AWS_CONTAINER_CREDENTIALS_RELATIVE_URI
```

L'output previsto è il seguente:

```
{
 "RoleArn": "arn:aws:iam::123456789012:role/SSMTaskRole-SSMFargateTaskIAMRole-
DASWWSF2WGD6",
 "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
 "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
 "Token": "IQoJb3JpZ2luX2VjEEM/Example==",
 "Expiration": "2021-01-16T00:51:53Z"
}
```

Le versioni più recenti delAWSGli SDK recuperano automaticamente queste credenziali dal`AWS_CONTAINER_CREDENTIALS_RELATIVE_URI`ambiente varable quando si effettuaAWSChiamate alle API.

L'output include una coppia di chiavi di accesso costituita da un ID chiave di accesso segreta e una chiave segreta utilizzata dall'applicazione per accedere aAWSrisorse AWS. Include anche un token cheAWSPer verificare che le credenziali siano valide. Per impostazione predefinita, le credenziali assegnate alle attività che utilizzano ruoli attività sono valide per sei ore. Successivamente, vengono automaticamente ruotati dall'agente del container Amazon ECS.

Ruolo per l'esecuzione di attività

Il ruolo di esecuzione dell'attività viene utilizzato per concedere all'agente contenitore Amazon ECS l'autorizzazione a chiamare specificiAWSAzioni API per conto dell'utente. Ad esempio, quando utilizziAWS Fargate, Fargate ha bisogno di un ruolo IAM che gli permetta di estrarre immagini da Amazon ECR e scrivere registri nei registri CloudWatch Logs. Un ruolo IAM è richiesto anche quando un'attività fa riferimento a un segreto archiviato inAWS Secrets Manager, ad esempio un'immagine pull secret.

> **Note**
>
> Se si estrae le immagini come utente autenticato, è meno probabile che si verifichino le modifiche apportate aLimiti della velocità di pull di Docker Hub: . Per ulteriori informazioni, consulta .Autenticazione di registri privati per istanze di container: .
> Utilizzando Amazon ECR e Amazon ECR Public, puoi evitare i limiti imposti da Docker. Se si estrae immagini da Amazon ECR, questo consente anche di ridurre i tempi di pull della rete e di ridurre le modifiche al trasferimento dei dati quando il traffico lascia il VPC.

> **Important**
>
> Quando si utilizza Fargate, è necessario autenticarsi in un registro di immagini private utilizzando`repositoryCredentials`: . Non è possibile impostare le variabili di ambiente dell'agente contenitore Amazon ECS`ECS_ENGINE_AUTH_TYPE`o`ECS_ENGINE_AUTH_DATA`o modificare`ecs.config`per le attività ospitate su Fargate. Per ulteriori informazioni, consultaAutenticazione di registri privati per attività: .

Ruolo dell'istanza del container Amazon EC2

L'agente contenitore Amazon ECS è un contenitore che viene eseguito su ogni istanza di Amazon EC2 in un cluster Amazon ECS. È inizializzato al di fuori di Amazon ECS utilizzando il metodo`init`disponibile nel sistema operativo. Di conseguenza, non possono essere concesse autorizzazioni tramite un ruolo di attività. Invece, le autorizzazioni devono essere assegnate alle istanze Amazon EC2 su cui vengono eseguiti gli agenti. L'elenco delle azioni nell'esempio`AmazonEC2ContainerServiceforEC2Role`la politica deve essere concessa a`ecsInstanceRole`: . Se non si esegue questa operazione, le istanze non potranno aderire al cluster.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeTags",
                "ecs:CreateCluster",
                "ecs:DeregisterContainerInstance",
                "ecs:DiscoverPollEndpoint",
                "ecs:Poll",
                "ecs:RegisterContainerInstance",
                "ecs:StartTelemetrySession",
                "ecs:UpdateContainerInstancesState",
                "ecs:Submit*",
                "ecr:GetAuthorizationToken",
                "ecr:BatchCheckLayerAvailability",
                "ecr:GetDownloadUrlForLayer",
                "ecr:BatchGetImage",
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": "*"
        }
    ]
}
```

In questo criterio, il`ecr`e`logs`consentono ai contenitori in esecuzione sulle istanze di estrarre immagini da Amazon ECR e scrivere registri su Amazon CloudWatch. La`ecs`consentono all'agente di

registrare e annullare la registrazione delle istanze e di comunicare con il piano di controllo Amazon ECS. Di questi, il`ecs:CreateCluster`è facoltativa.

Ruoli collegati ai servizi

Puoi utilizzare il ruolo collegato ai servizi affinché Amazon ECS conceda al servizio Amazon ECS l'autorizzazione a chiamare altre API di servizio per tuo conto. Amazon ECS ha bisogno delle autorizzazioni per creare ed eliminare interfacce di rete, registrare e annullare la registrazione degli obiettivi con un gruppo target. Inoltre, richiede le autorizzazioni necessarie per creare ed eliminare policy di ridimensionamento. Queste autorizzazioni vengono concesse tramite il ruolo collegato ai servizi. Questo ruolo viene creato per conto dell'utente la prima volta che si utilizza il servizio.

> **Note**
>
> Se elimini inavvertitamente il ruolo collegato ai servizi, puoi ricrearlo. Per istruzioni, consultaCreazione del ruolo collegato ai servizi: .

Recommendations

Ti consigliamo anche di completare le seguenti operazioni durante la configurazione di ruoli e policy IAM.

Blocca l'accesso ai metadati Amazon EC2

Quando esegui le tue attività su istanze Amazon EC2, ti consigliamo vivamente di bloccare l'accesso ai metadati Amazon EC2 per evitare che i contenitori ereditino il ruolo assegnato a tali istanze. Se le tue applicazioni devono chiamare unAWS, utilizzare invece i ruoli IAM per le attività.

Per impedire l'esecuzione di attività inpontedi accedere ai metadati Amazon EC2, eseguire il seguente comando o aggiornare i dati utente dell'istanza. Per ulteriori istruzioni sull'aggiornamento dei dati utente di un'istanza, vedere questoAWSArticolo di Support: . Per ulteriori informazioni sulla modalità bridge di definizione delle attività, consulta.modalità di rete definizione attività: .

```
sudo yum install -y iptables-services; sudo iptables --insert FORWARD 1 --in-interface
 docker+ --destination 192.0.2.0/32 --jump DROP
```

Affinché questa modifica continui dopo un riavvio, esegui il seguente comando specifico per Amazon Machine Image (AMI):

- Amazon Linux 2

```
sudo iptables-save | sudo tee /etc/sysconfig/iptables && sudo systemctl enable --now
 iptables
```

- Amazon Linux

```
sudo service iptables save
```

Per le attività che utilizzanoawsvpcModalità di rete, impostare la variabile d'ambienteECS_AWSVPC_BLOCK_IMDSDa `atrue`nella`/etc/ecs/ecs.config`file.

È necessario impostare la proprietàECS_ENABLE_TASK_IAM_ROLE_NETWORK_HOSTVariabile di`false`nel file di configurazione ecs-agent per impedire che i contenitori in esecuzione all'interno del file`host`dall'accesso ai metadati Amazon EC2.

Utilizza**awsvpc**Modalità di rete

Usa la reteawsvpcper limitare il flusso di traffico tra diverse attività o tra le tue attività e altri servizi eseguiti all'interno del tuo VPC Amazon. Questo aggiunge un ulteriore livello di sicurezza. La`awsvpc`fornisce l'isolamento della rete a livello di attività per le attività eseguite su Amazon EC2. È la modalità predefinita inAWS Fargate. è l'unica modalità di rete che è possibile utilizzare per assegnare un gruppo di sicurezza alle attività.

Utilizzare IAM Access Advisor per perfezionare i ruoli

Ti consigliamo di rimuovere tutte le azioni che non sono mai state utilizzate o che non sono state utilizzate per un certo periodo di tempo. Ciò impedisce l'accesso indesiderato. A tale scopo, esaminare i risultati prodotti da IAM Access Advisor e quindi rimuovere le azioni che non sono mai state utilizzate o non sono state utilizzate di recente. A tale scopo, attenersi alla procedura descritta di seguito.

Eseguire il comando seguente per generare un report che mostra le ultime informazioni di accesso per il criterio di riferimento:

```
aws iam generate-service-last-accessed-details --arn arn:aws:iam::123456789012:policy/
ExamplePolicy1
```

Utilizzo dell'`JobId`che era nell'output per eseguire il comando seguente. A quel punto è possibile visualizzare i risultati del report.

```
aws iam get-service-last-accessed-details --job-id 98a765b4-3cde-2101-2345-example678f9
```

Per ulteriori informazioni, consultaIAM Access Advisor: .

MonitorareAWS CloudTrailPer attività sospette

È possibile monitorareAWS CloudTrailper qualsiasi attività sospetta. MostAWSLe chiamate API vengono registrate inAWS CloudTrailCome eventi. Sono analizzati daAWS CloudTrailApprofondimenti e vieni avvisato di eventuali comportamenti sospetti associati a`write`Chiamate alle API. Ciò potrebbe includere un picco nel volume delle chiamate. Questi avvisi includono informazioni quali l'ora in cui si è verificata l'attività insolita e l'ARN di identità superiore che ha contribuito alle API.

È possibile identificare le azioni eseguite da attività con un ruolo IAM inAWS CloudTrailguardando il`userIdentity`proprietà. Nell'esempio seguente, la`arn`include il nome del ruolo assunto,`s3-write-go-bucket-role`, seguito dal nome dell'attività,`7e9894e088ad416eb5cab92afExample`: .

```
"userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROA36C6WWEJ2YEXAMPLE:7e9894e088ad416eb5cab92afExample",
    "arn": "arn:aws:sts::123456789012:assumed-role/s3-write-go-bucket-
role/7e9894e088ad416eb5cab92afExample",
    ...
}
```

> **Note**
>
> Quando le attività che assumono un ruolo vengono eseguite su istanze di container Amazon EC2, l'agente contenitore Amazon ECS registra una richiesta nel registro di controllo dell'agente che si trova in un indirizzo nel`/var/log/ecs/audit.log.YYYY-MM-DD-HH`. Per ulteriori informazioni, consultaRegistro ruoli IAM attivitàeRegistrazione di eventi Insights per i trail: .

Sicurezza di rete

La sicurezza di rete è un argomento ampio che comprende diversi sottoargomenti. Questi includono la crittografia in transito, la segmentazione e l'isolamento della rete, il firewall, il routing del traffico e l'osservabilità.

Crittografia in transito

La crittografia del traffico di rete impedisce agli utenti non autorizzati di intercettare e leggere i dati quando tali dati vengono trasmessi attraverso una rete. Con Amazon ECS, la crittografia di rete può essere implementata in uno dei modi seguenti.

- Con una rete di servizio (TLS):

  conAWS App Mesh, è possibile configurare le connessioni TLS tra i proxy inviati distribuiti con endpoint mesh. Due esempi sono i nodi virtuali e i gateway virtuali. I certificati TLS possono provenire daAWS Certificate Manager(ACM). In alternativa, può provenire dalla propria autorità di certificazione privata.

  - Abilitazione della sicurezza Transport Layer (TLS)
  - Abilitare la crittografia del traffico tra i servizi inAWS App Meshutilizzando certificati ACM o certificati forniti dal cliente
  - Procedura guidata per TLS ACM
  - Procedura guidata per i file TLS
  - Envoy
- Utilizzo delle istanze Nitro:

  Per impostazione predefinita, il traffico viene crittografato automaticamente tra i seguenti tipi di istanza Nitro: C5n, G4, I3en, M5dn, M5dn, P3dn, P3dn, P3dn e R5n. Il traffico non viene crittografato quando viene instradato attraverso un gateway di transito, un servizio di bilanciamento del carico o un intermediario simile.

  - Crittografia in transito
  - Novità del conto dal 2019
  - Questo discorso da Re:inforce 2019
- Utilizzare la Server Name Indication (SNI) con un sistema di Application Load Balancer:

Application Load Balancer (ALB) e Network Load Balancer (NLB) supportano Server Name Indication (SNI). Utilizzando SNI, è possibile mettere più applicazioni sicure dietro un singolo listener. Per questo, ognuno ha il proprio certificato TLS. Si consiglia di eseguire il provisioning dei certificati per il load balanacer utilizzandoAWS Certificate Manager(ACM) e quindi aggiungerli all'elenco dei certificati del listener. LaAWSIl sistema di bilanciamento del carico utilizza un algoritmo intelligente di selezione dei certificati con SNI. Se il nome host fornito da un client corrisponde a un singolo certificato nell'elenco dei certificati, il sistema di bilanciamento del carico sceglie tale certificato. Se un nome host fornito da un client corrisponde a più certificati nell'elenco, il sistema di bilanciamento del carico seleziona un certificato che il client è in grado di supportare. Esempi includono un certificato autofirmato o un certificato generato tramite l'ACM.

- SNI con Application Load Balancer
- SNI con Network Load Balancer

- Crittografia end-to-end con certificati TLS:

  Ciò comporta la distribuzione di un certificato TLS con l'attività. Può trattarsi di un certificato autofirmato o di un certificato da un'autorità di certificazione attendibile. È possibile ottenere il certificato facendo riferimento a un segreto per il certificato. In caso contrario, è possibile scegliere di eseguire un contenitore che emette una richiesta di firma del certificato (CSR) a ACM e quindi monta il segreto risultante in un volume condiviso.

  - Mantenere la sicurezza del livello di trasporto fino ai container utilizzando Network Load Balancer con Amazon ECS parte 1
  - Mantenere Transport Layer Security (TLS) fino alla parte 2 del container: Uso di AWS Private Certificate Authority

Reti di attività

I seguenti consigli sono in considerazione del funzionamento di Amazon ECS. Amazon ECS non utilizza una rete di sovrapposizione. Invece, le attività sono configurate per funzionare in diverse modalità di rete. Ad esempio, le attività configurate per utilizzare`bridge`acquisisce un indirizzo IP non instradabile da una rete Docker in esecuzione su ciascun host. Attività configurate per l'utilizzo del comando`awsvpc`Acquisisce un indirizzo IP dalla sottorete dell'host. Attività configurate con`host`utilizzano l'interfaccia di rete dell'host.`awsvpc`è la modalità di rete preferita. Questo perché è l'unica modalità che è possibile utilizzare per assegnare gruppi di sicurezza alle attività. È anche l'unica modalità disponibile perAWS Fargateattività su Amazon ECS.

Gruppi di sicurezza per le attività

Ti consigliamo di configurare le attività in modo da utilizzare la`awsvpc`Modalità di rete. Dopo aver configurato l'attività per l'utilizzo di questa modalità, l'agente Amazon ECS esegue automaticamente il provisioning e allega un'interfaccia di rete elastica (ENI) all'attività. Quando viene eseguito il provisioning dell'ENI, l'attività viene registrata in unAWSgruppo di sicurezza. Il gruppo di sicurezza agisce da firewall virtuale che è possibile utilizzare per controllare il traffico in entrata e in uscita.

Rete di servizio e MTL (Mutual Transport Layer Security)

È possibile utilizzare una mesh di servizio comeAWS App Meshper controllare il traffico di rete. Per impostazione predefinita, un nodo virtuale può comunicare solo con i relativi back-end di servizio configurati, ad esempio i servizi virtuali con cui il nodo virtuale comunicherà. Se un nodo virtuale deve comunicare con un servizio esterno alla mesh, è possibile utilizzare il comando`ALLOW_ALL`filtro in uscita o creando un nodo virtuale all'interno della mesh per il servizio esterno. Per ulteriori informazioni, consultaProcedura dettagliata sull'uscita Kubernetes: .

App Mesh offre inoltre la possibilità di utilizzare Mutual Transport Layer Security (MTL) in cui sia il client che il server vengono autenticati reciprocamente utilizzando i certificati. La successiva comunicazione tra client e server viene quindi crittografata utilizzando TLS. Richiedendo MTL tra i servizi in una mesh, è possibile verificare che il traffico provenga da un'origine attendibile. Per ulteriori informazioni, consultare i seguenti argomenti:

- Autenticazione MTL
- Procedura dettagliata per MTLS Secret Discovery Service (SDS)
- Procedura guidata per i file MTLS

AWS PrivateLink

AWS PrivateLinkè una tecnologia di rete che consente di creare endpoint privati per diversiAWSservizi, tra cui Amazon ECS. Gli endpoint sono necessari in ambienti sandbox in cui non sono collegati Internet Gateway (IGW) al VPC di Amazon e non sono presenti percorsi alternativi a Internet. Utilizzo diAWS PrivateLinkassicura che le chiamate al servizio Amazon ECS rimangano all'interno del VPC Amazon e non attraversino Internet. Per istruzioni su come creareAWS PrivateLinkper Amazon ECS e altri servizi correlati, vedereInterfaccia Amazon ECS Endpoint VPC di Amazon: .

> **Important**
>
> AWS Fargatele attività non richiedono unAWS PrivateLinkendpoint per Amazon ECS.

Amazon ECR e Amazon ECS supportano entrambe le politiche degli endpoint. Questi criteri consentono di perfezionare l'accesso alle API di un servizio. Ad esempio, puoi creare una politica endpoint per Amazon ECR che consenta solo il push delle immagini nei registri, in particolareAWSaccount. Una politica come questa potrebbe essere utilizzata per evitare che i dati vengano esfiltrati attraverso le immagini del contenitore, pur consentendo agli utenti di inviare il push ai registri ECR Amazon autorizzati. Per ulteriori informazioni, consultaUtilizzare i criteri di endpoint VPC: .

La policy seguente permette tutte leAWSentità nel tuo account per eseguire tutte le azioni nei confronti solo dei tuoi repository ECR Amazon:

```
{
  "Statement": [
    {
      "Sid": "LimitECRAccess",
      "Principal": "*",
      "Action": "*",
      "Effect": "Allow",
      "Resource": "arn:aws:ecr:region:your_account_id:repository/*"
    },
  ]
}
```

È possibile migliorare ulteriormente questa impostazione impostando una condizione che utilizza il nuovo`PrincipalOrgID`proprietà. Ciò impedisce il push e il pull delle immagini da parte di un principal IAM che non fa parte del tuoAWS Organizations: . Per ulteriori informazioni, consultaaws:PrincipalOrgID: .

Si consiglia di applicare lo stesso criterio sia `acom.amazonaws.region.ecr.dkr`e la`com.amazonaws.region.ecr.api`Endpoint .

Impostazioni dell'agente del container Amazon ECS

Il file di configurazione dell'agente contenitore Amazon ECS include diverse variabili di ambiente relative alla sicurezza della

rete.`ECS_AWSVPC_BLOCK_IMDS`e`ECS_ENABLE_TASK_IAM_ROLE_NETWORK_HOST`vengono utilizzati per bloccare l'accesso di un'attività ai metadati Amazon EC2.`HTTP_PROXY`viene utilizzato per configurare l'agente per instradare attraverso un proxy HTTP per connettersi a Internet. Per istruzioni sulla configurazione dell'agente e del runtime Docker affinché instradino attraverso un proxy, vedereConfigurazione di un proxy HTTP: .

> **Important**
>
> Queste impostazioni non sono disponibili quando si utilizzaAWS Fargate: .

Recommendations

Ti consigliamo di effettuare le seguenti operazioni durante la configurazione del VPC, dei bilanciamenti del carico e della rete Amazon.

Utilizzare la crittografia di rete, ove applicabile

È consigliabile utilizzare la crittografia di rete, ove applicabile. Alcuni programmi di conformità, ad esempio PCI DSS, richiedono la crittografia dei dati in transito se i dati contengono dati del titolare della carta. Se il carico di lavoro ha requisiti simili, configurare la crittografia di rete.

I browser moderni avvisano gli utenti quando si connettono a siti non sicuri. Se il servizio è gestito da un servizio di bilanciamento del carico pubblico, utilizzare TLS/SSL per crittografare il traffico dal browser del client al servizio di bilanciamento del carico e, se necessario, eseguire nuovamente la crittografia al back-end.

Utilizza**`awsvpc`**modalità di rete e gruppi di protezione quando è necessario controllare il traffico tra attività o tra attività e altre risorse di rete

È consigliabile utilizzare`awsvpc`modalità di rete e gruppi di protezione quando è necessario controllare il traffico tra attività e tra attività e altre risorse di rete. Se il servizio è dietro un ALB, utilizzare i gruppi di sicurezza per consentire solo il traffico in ingresso proveniente da altre risorse di rete utilizzando lo stesso gruppo di protezione dell'ALB. Se la tua applicazione è dietro un Bilanciamento carico di rete, configura il gruppo di sicurezza dell'attività in modo da consentire solo il traffico in entrata dall'intervallo CIDR VPC di Amazon e gli indirizzi IP statici assegnati al Bilanciamento carico di rete.

I gruppi di sicurezza devono essere utilizzati anche per controllare il traffico tra le attività e altre risorse all'interno del VPC Amazon, ad esempio i database RDS di Amazon.

Creare cluster in vPC Amazon separati quando il traffico di rete deve essere rigorosamente isolato

Dovresti creare cluster in vPC Amazon separati quando il traffico di rete deve essere rigorosamente isolato. Evitare l'esecuzione di carichi di lavoro con requisiti di sicurezza rigorosi nei cluster con carichi di lavoro che non devono rispettare tali requisiti. Quando è obbligatorio un isolamento di rete rigoroso, creare cluster in vPC Amazon separati ed esporre selettivamente i servizi ad altri vPC Amazon utilizzando gli endpoint Amazon VPC. Per ulteriori informazioni, consultaEndpoint Amazon VPC: .

ConfiguraAWS PrivateLinkendpoint se garantiti

È consigliabile configurareAWS PrivateLinkendpoint quando giustificato. Se la tua politica di sicurezza ti impedisce di collegare un Internet Gateway (IGW) ai tuoi vPC Amazon, configuraAWS PrivateLinkendpoint per Amazon ECS e altri servizi come Amazon ECR,AWS Secrets Managere Amazon CloudWatch.

Utilizzare i log di flusso di Amazon VPC per analizzare il traffico da e verso attività di lunga durata

È consigliabile utilizzare i log di flusso di Amazon VPC per analizzare il traffico da e verso attività di lunga durata. Attività che utilizzano`awsvpc`modalità di rete ottenere il proprio ENI. In questo modo, puoi monitorare il traffico che va da e verso singole attività utilizzando i log di flusso di Amazon VPC. Un recente aggiornamento ai log di flusso di Amazon VPC (v3), arricchisce i log con i metadati del traffico, tra cui l'ID vpc, l'ID della subnet e l'ID dell'istanza. Questi metadati possono essere utilizzati per restringere un'indagine. Per ulteriori informazioni, consultaLog di flusso Amazon VPC: .

> **Note**
>
> A causa della natura temporanea dei contenitori, i registri di flusso potrebbero non essere sempre un modo efficace per analizzare i modelli di traffico tra contenitori o contenitori diversi e altre risorse di rete.

Gestione dei segreti

I segreti, come le chiavi API e le credenziali del database, vengono spesso utilizzati dalle applicazioni per accedere ad altri sistemi. Spesso consistono in un nome utente e una password, un certificato o una chiave API. L'accesso a questi segreti dovrebbe essere limitato a entità IAM specifiche che utilizzano IAM e iniettate in contenitori in fase di esecuzione.

I segreti possono essere iniettati senza soluzione di continuità nei contenitori daAWS Secrets ManagerAmazon EC2 Systems Manager Parameter Store. Questi segreti possono essere referenziati nell'attività come uno dei seguenti.

1. Vengono referenziati come variabili di ambiente che utilizzano il metodo`secrets`Parametro di definizione del container
2. Sono referenziati come`secretOptions`se la piattaforma di registrazione richiede l'autenticazione. Per ulteriori informazioni, consultaOpzioni di configurazione della registrazione: .
3. Sono referenziati come segreti tirati da immagini che usano il`repositoryCredentials`parametro di definizione del contenitore se il Registro di sistema da cui viene estratto il contenitore richiede l'autenticazione. Utilizzare questo metodo per estrarre immagini da Docker Hub. Per ulteriori informazioni, consultaAutenticazione di registri privati per attività: .

Recommendations

Si consiglia di eseguire le seguenti operazioni durante la configurazione della gestione dei segreti.

UtilizzaAWS Secrets ManagerAmazon EC2 Systems Manager Parameter Store per archiviare materiali segreti

È necessario archiviare in modo sicuro le chiavi API, le credenziali del database e altri materiali segreti inAWS Secrets Managero come parametro crittografato nell'Amazon EC2 Systems Manager Parameter Store. Questi servizi sono simili perché sono entrambi archivi chiave-valore gestiti che utilizzanoAWS KMSPer crittografare i dati sensibili.AWS Secrets Manager, tuttavia, include anche la possibilità di ruotare automaticamente i segreti, generare segreti casuali e condividere segreti attraversoAWSaccount. Se ritieni queste caratteristiche importanti, usaAWS Secrets Manageraltrimenti utilizzare parametri crittografati.

> **Note**
>
> Attività che fanno riferimento a un segreto daAWS Secrets Managero Amazon EC2 Systems Manager Parameter Store richiedono unRuolo per l'esecuzione di attivitàcon una politica che concede ad Amazon ECS l'accesso al segreto desiderato e, se applicabile, alAWS KMSutilizzata per crittografare e decrittografare quel segreto.

> **Important**
>
> I segreti a cui si fa riferimento nelle attività non vengono ruotati automaticamente. Se il segreto cambia, è necessario forzare una nuova distribuzione o avviare una nuova attività per recuperare il valore segreto più recente. Per ulteriori informazioni, consultare i seguenti argomenti:
>
> - AWS Secrets Manager: Iniezione di dati come variabili di ambiente
> - Amazon EC2 Systems Manager Parameter Store: Iniezione di dati come variabili di ambiente

Recupero di dati da un bucket Amazon S3 crittografato

Poiché il valore delle variabili di ambiente può inavvertitamente perdere nei registri e viene rivelato durante l'esecuzione`docker inspect`, dovresti archiviare i segreti in un bucket Amazon S3 crittografato e utilizzare i ruoli delle attività per limitare l'accesso a tali segreti. Quando si esegue questa operazione, l'applicazione deve essere scritta per leggere il segreto dal bucket Amazon S3. Per istruzioni, consultaImpostazione del comportamento di crittografia lato server predefinito per i bucket Amazon S3: .

Montare il segreto su un volume utilizzando un contenitore sidecar

Poiché esiste un rischio elevato di perdita di dati con le variabili di ambiente, è necessario eseguire un contenitore sidecar che legga i segreti daAWS Secrets Managere scriverli in un volume condiviso. Questo contenitore può essere eseguito e uscire prima del contenitore dell'applicazione utilizzandoOrdinamento del container Amazon ECS: . Quando si esegue questa operazione, il contenitore dell'applicazione viene successivamente montato il volume in cui è stato scritto il segreto. Come il metodo bucket Amazon S3, la tua applicazione deve essere scritta per leggere il segreto dal volume condiviso. Poiché il volume è ricoperto dall'ambito dell'attività, il volume viene eliminato

automaticamente dopo l'interruzione dell'attività. Per un esempio di contenitore sidecar, vedere la sezioneaws-secret-sidecar-iniettoreProgetto.

> **Note**
>
> Su Amazon EC2, il volume su cui è scritto il segreto può essere crittografato con unAWS KMSchiave gestita dal cliente. SuAWS Fargate, l'archiviazione dei volumi viene crittografata automaticamente utilizzando una chiave gestita dal servizio.

Altre risorse

- Passare segreti ai container in un'attività Amazon ECS
- Cameraè un wrapper per la memorizzazione di segreti nell'Amazon EC2 Systems Manager Parameter Store

Compliance

La tua responsabilità di conformità durante l'utilizzo di Amazon ECS è determinata dalla riservatezza dei dati, dagli obiettivi di conformità dell'azienda e dalle leggi e normative in vigore.

AWSFornisce le seguenti risorse per facilitare la conformità:

- Guide di avvio rapido per la sicurezza e la conformità: Queste guide alla distribuzione illustrano considerazioni relative all'architettura e forniscono procedure per la distribuzione di ambienti di base incentrati sulla sicurezza e sulla conformità suAWS: .
- Whitepaper Architecting for HIPAA Security and Compliance: Questo white paper descrive come le aziende possono utilizzareAWSPer creare applicazioni conformi a HIPAA.
- AWSServizi nell'ambito del programma di compliance: Questo elenco contiene laAWSServizi nell'ambito di specifici programmi di conformità. Per ulteriori informazioni, consultaAWSProgramma di conformità: .

Payment Card Industry Data Security Standard (PCI DSS)

È importante comprendere il flusso completo dei dati del titolare della carta (CHD) all'interno dell'ambiente quando si aderisce a PCI DSS. Il flusso CHD determina l'applicabilità del DSS PCI,

definisce i confini e i componenti di un ambiente dati titolare di carte (CDE) e quindi l'ambito di una valutazione PCI DSS. La determinazione accurata dell'ambito PCI DSS è fondamentale per definire la posizione di sicurezza e, in ultima analisi, una valutazione di successo. I clienti devono disporre di una procedura per la determinazione dell'ambito che ne garantisca la completezza e rilevi modifiche o deviazioni dall'ambito.

La natura temporanea delle applicazioni containerizzate fornisce ulteriori complessità durante il controllo delle configurazioni. Di conseguenza, i clienti devono essere consapevoli di tutti i parametri di configurazione dei container per garantire che i requisiti di conformità vengano risolti in tutte le fasi del ciclo di vita dei container.

Per ulteriori informazioni su come ottenere la conformità PCI DSS su Amazon ECS, consulta i seguenti white paper.

- Architecting on Amazon ECS per conformità PCI DSS
- Architettura per l'ambito e la segmentazione PCI DSS su AWS

HIPAA (Legge Health ability and Accountability Act)

L'utilizzo di Amazon ECS con carichi di lavoro che elaborano informazioni sanitarie protette (PHI) non richiede alcuna configurazione aggiuntiva. Amazon ECS funge da servizio di orchestrazione che coordina il lancio di container su Amazon EC2. Non funziona con o su dati all'interno del carico di lavoro orchestrato. Conformemente alle normative HIPAA e alleAWSBusiness Associate Addendum, PHI deve essere crittografato in transito e in stato di inattività quando si accede ai container lanciati con Amazon ECS.

Vari meccanismi per la crittografia a riposo sono disponibili con ogniAWS, ad esempio Amazon S3, Amazon EBS eAWS KMS: . È possibile distribuire una rete overlay (come VNS3 o Weave Net) per garantire la crittografia completa del PHI trasferito tra contenitori o per fornire un livello ridondante di crittografia. Anche la registrazione completa deve essere abilitata e tutti i log dei contenitori devono essere indirizzati ad Amazon CloudWatch. Per ulteriori informazioni, consultaArchitecting for HIPAA Security and Compliance: .

Recommendations

È necessario coinvolgere in anticipo i proprietari del programma di conformità all'interno della propria azienda e utilizzare ilAWSModello di responsabilità condivisaper identificare la proprietà del controllo di conformità per il successo con i programmi di conformità pertinenti.

Logging e monitoraggio

La registrazione e il monitoraggio sono importanti per mantenere l'affidabilità, la disponibilità e le prestazioni di Amazon ECS e deiAWSSoluzioni.AWSfornisce diversi strumenti per il monitoraggio delle risorse Amazon ECS e la risposta a potenziali incidenti:

- Allarmi Amazon CloudWatch
- Amazon CloudWatch Logs
- Amazon CloudWatch Events
- AWS CloudTrail Log

Puoi configurare i container nelle attività affinché inviino informazioni di log ad Amazon CloudWatch Logs. Se usiAWS FargatePer la tua attività, puoi visualizzare i log provenienti dai tuoi container. Se utilizzi il tipo di avvio Amazon EC2, puoi visualizzare diversi log dei container in un'unica comoda ubicazione. In questo modo viene inoltre impedito che i log del container occupino spazio su disco nelle istanze di container.

Per ulteriori informazioni su Amazon CloudWatch Logs, consulta.Monitorare i log dalle istanze Amazon EC2nellaGuida per l'utente di Amazon CloudWatch: . Per istruzioni sull'invio di log del container dalle attività ad Amazon CloudWatch Logs, consultaUtilizzo `diawslogs`driver di registro: .

Registrazione del contenitore con Fluent Bit

AWSfornisce un'immagine Fluent Bit con plugin per Amazon CloudWatch Logs e Amazon Kinesis Data Firehose. Questa immagine fornisce la possibilità di instradare i registri alle destinazioni Amazon CloudWatch e Amazon Kinesis Data Firehose (che includono Amazon S3, Amazon Elasticsearch Service e Amazon Redshift). Si consiglia di utilizzare Fluent Bit come router di log perché dispone di un tasso di utilizzo delle risorse inferiore a Fluentd. Per ulteriori informazioni, consultaAmazon CloudWatch Logs per Fluent BiteAmazon Kinesis Data Firehose per Fluent Bit: .

LaAWSPer l'immagine Fluent Bit è disponibile su:

- Amazon ECR su Amazon ECR Public Gallery
- Repository Amazon ECR(nella maggior parte delle regioni ad alta disponibilità)
- Docker Hub

Di seguito è mostrata la sintassi da utilizzare per l'interfaccia della riga di comando Docker.

```
docker pull public.ecr.aws/aws-observability/aws-for-fluent-bit:tag
```

Ad esempio, è possibile estrarre l'ultimoAWSper l'immagine Fluent Bit utilizzando questo comando CLI Docker:

```
docker pull public.ecr.aws/aws-observability/aws-for-fluent-bit:latest
```

Leggi inoltre i seguenti post del blog per ulteriori informazioni su Fluent Bit e sulle funzionalità correlate:

- Bit fluente per Amazon EKS su AWS Fargate
- Registrazione centralizzata dei container con Fluent Bit
- Creazione di un aggregatore di soluzioni di log scalabile conAWS Fargate, Fluentd e Amazon Kinesis Data Firehose

Routing dei registri personalizzato - FireLens per Amazon ECS

Con FireLens per Amazon ECS, puoi utilizzare i parametri di definizione dell'attività per instradare i log a unAWSService oAWSDestinazione APN (Partner Network) per lo storage e l'analisi dei log. FireLens funziona con Fluentd e Fluent Bit. Forniamo ilAWSper l'immagine Fluent Bit. In alternativa, puoi utilizzare la tua immagine Fluentd o Fluent Bit.

Quando utilizzi FireLens per Amazon ECS, dovresti prendere in considerazione le seguenti condizioni e considerazioni:

- FireLens per Amazon ECS è supportato per le attività ospitate sia suAWS Fargatee Amazon EC2.
- FireLens per Amazon ECS è supportato inAWS CloudFormationModelli di. Per ulteriori informazioni, consultaAWS::ECS::TaskDefinition FirelensConfigurationnellaAWS CloudFormationGuida per l'utente: .
- Per le attività che utilizzano il`bridge`In modalità di rete, i container con la configurazione FireLens devono essere avviati prima dell'avvio di qualsiasi container dell'applicazione che si basa su di esso. Per controllare l'ordine di avvio dei container, utilizza le condizioni di dipendenza nella definizione dell'attività. Per ulteriori informazioni, consultaDipendenze per: .

Sicurezza di AWS Fargate

Ti consigliamo anche di tenere in considerazione le seguenti best practice quando utilizziAWS Fargate: .

UtilizzaAWS KMSper crittografare l'archiviazione effimera

Dovresti avere il tuo archivio effimero crittografato daAWS KMS: . Per le attività Amazon ECS ospitate suAWS FargateUtilizzo della versione della piattaforma`1.4.0`o versioni successive, ciascuna attività riceve 20 GB di memoria effimera. La quantità di storage non è regolabile. Per tali attività lanciate il 28 maggio 2020 o successivamente, lo storage temporaneo viene crittografato con un algoritmo di crittografia AES-256 utilizzando una chiave di crittografia gestita daAWS Fargate: .

Esempio: Avvio di un'attività Amazon ECS suAWS FargatePiattaforma versione 1.4.0 con crittografia di archiviazione effimera

Il comando seguente avvierà un'attività Amazon ECS suAWS Fargateversione 1.4 della piattaforma. Poiché questa attività viene avviata come parte del cluster Amazon ECS, utilizza 20 GB di memoria effimera crittografata automaticamente.

```
aws ecs run-task --cluster clustername \
  --task-definition taskdefinition:version \
  --count 1
  --launch-type "FARGATE" \
  --platform-version 1.4.0 \
  --network-configuration
 "awsvpcConfiguration={subnets=[subnetid],securityGroups=[securitygroupid]}" \
  --region region
```

Funzionalità SYS_PTRACE per il trace syscall del kernel

La configurazione predefinita delle funzionalità Linux aggiunte o rimosse dal contenitore viene fornita da Docker. Per ulteriori informazioni sulle funzionalità disponibili, consulta.Privilegio runtime e funzionalità LinuxnellaEsecuzione di Dockerdocumentazione.

Attività avviate suAWS Fargatesupporta solo l'aggiunta di`SYS_PTRACE`Capacità del kernel.

Fare riferimento al video tutorial qui sotto che mostra come utilizzare questa funzione tramite SysdigFalcoProgetto.

#ContainersFromTheCouch - Risoluzione dei problemi relativi allaAWS FargateAttività che utilizza la funzionalità SYS_PTRACE

Il codice discusso nel video precedente può essere trovato su GitHubQui: .

Sicurezza di attività e container

Dovresti considerare l'immagine del container come la tua prima linea di difesa contro un attacco. Un'immagine insicura e mal costruita può consentire a un utente malintenzionato di sfuggire ai limiti del contenitore e ottenere l'accesso all'host. Dovresti fare quanto segue per mitigare il rischio che ciò accada.

Recommendations

Ti consigliamo anche di completare le seguenti operazioni durante la configurazione di container e attività.

Creare immagini minime o utilizzare immagini distroless

Iniziare rimuovendo tutti i file binari estranei dall'immagine contenitore. Se si utilizza un'immagine sconosciuta da Docker Hub, controllare l'immagine per fare riferimento al contenuto di ciascuno dei livelli del contenitore. È possibile utilizzare un'applicazione comeImmersionePer farlo.

In alternativa, puoi utilizzaresenza distrolessimmagini che includono solo l'applicazione e le relative dipendenze di runtime. Non contengono gestori di pacchetti o shell. Le immagini senza distroless migliorano il «segnale al rumore degli scanner e riducono l'onere di stabilire la provenienza esattamente ciò di cui hai bisogno». Per ulteriori informazioni, consulta la documentazione di GitHub susenza distroless: .

Docker ha un meccanismo per creare immagini da un'immagine riservata e minima nota comegraffio: . Informazioni su Formore, consultaCreazione di una semplice immagine genitore utilizzandograffionella documentazione Docker. Con langage come Go, puoi creare un binario collegato statico e fare riferimento al tuo Dockerfile. L'esempio seguente mostra come puoi eseguire questa operazione.

```
############################
# STEP 1 build executable binary
############################
```

```
FROM golang:alpine AS builder
# Install git.
# Git is required for fetching the dependencies.
RUN apk update && apk add --no-cache git
WORKDIR $GOPATH/src/mypackage/myapp/
COPY . .
# Fetch dependencies.
# Using go get.
RUN go get -d -v
# Build the binary.
RUN go build -o /go/bin/hello
############################
# STEP 2 build a small image
############################
FROM scratch
# Copy our static executable.
COPY --from=builder /go/bin/hello /go/bin/hello
# Run the hello binary.
ENTRYPOINT ["/go/bin/hello"]
This creates a container image that consists of your application and nothing else,
 making it extremely secure.
```

L'esempio precedente è anche un esempio di una build a più stadi. Questi tipi di build sono interessanti dal punto di vista della sicurezza perché è possibile utilizzarli per ridurre al minimo le dimensioni dell'immagine finale inviata al Registro di sistema contenitore. Le immagini del contenitore prive di strumenti di costruzione e di altri file binari estranei migliorano la posizione di sicurezza riducendo la superficie di attacco dell'immagine. Per ulteriori informazioni sulle build in più fasi, consultacreazione di build multistadio: .

Scansiona le tue immagini alla ricerca di vulnerabilità

Analogamente alle controparti delle macchine virtuali, le immagini dei contenitori possono contenere file binari e librerie di applicazioni con vulnerabilità o sviluppare vulnerabilità nel tempo. Il modo migliore per salvaguardare gli exploit è scansionare regolarmente le immagini con uno scanner di immagini. Le immagini memorizzate in Amazon ECR possono essere scansionate su push o on-demand (una volta ogni 24 ore). Al momento l'ECR di Amazon utilizzaClair, una soluzione open source di scansione delle immagini. Dopo la scansione di un'immagine, i risultati vengono registrati nel flusso di eventi Amazon ECR in Amazon EventBridge. Puoi anche visualizzare i risultati di una scansione dalla console Amazon ECR o chiamando ilDescribeImageScanFallingsAPI. Immagini con`HIGHoCRITICAL`deve essere eliminata o ricostruita. Se un'immagine distribuita sviluppa una vulnerabilità, deve essere sostituita il prima possibile.

Docker Desktop Edge versione 2.3.6.0o versione successiva puòscanimmagini locali. Le scansioni sono alimentate daSnyk, un servizio di sicurezza delle applicazioni. Quando vengono rilevate vulnerabilità, Snyk identificherà i livelli e le dipendenze con la vulnerabilità nel Dockerfile. Raccomanda anche alternative sicure come l'utilizzo di un'immagine di base più sottile con meno vulnerabilità o l'aggiornamento di un particolare pacchetto a una versione più recente. Utilizzando Docker scan, gli sviluppatori possono risolvere potenziali problemi di sicurezza prima di inviare le immagini nel Registro di sistema.

- Automatizzare la conformità delle immagini utilizzando Amazon ECR eAWS Security Hubspiega come far emergere le informazioni sulla vulnerabilità da Amazon ECR inAWS Security Hube automatizzare la correzione bloccando l'accesso alle immagini vulnerabili.

Rimuovi autorizzazioni speciali dalle tue immagini

I flag dei diritti di accesso`setuid`e`setgid`consentono l'esecuzione di un eseguibile con le autorizzazioni del proprietario o del gruppo dell'eseguibile. Rimuovere tutti i file binari con questi diritti di accesso dall'immagine poiché questi binari possono essere utilizzati per aumentare i privilegi. Considera la rimozione di tutte le shell e le utilità come`nc`e`curl`che può essere utilizzato per scopi dannosi. Puoi trovare i file con`setuid`e`setgid`Accedere mediante il comando riportato di seguito.

```
find / -perm /6000 -type f -exec ls -ld {} \;
```

Per rimuovere queste autorizzazioni speciali da questi file, aggiungere la seguente direttiva all'immagine contenitore.

```
RUN find / -xdev -perm /6000 -type f -exec chmod a-s {} \; || true
```

Creare un insieme di immagini curate

Anziché consentire agli sviluppatori di creare le proprie immagini, creare una serie di immagini controllate per i diversi stack di applicazioni nell'organizzazione. In questo modo, gli sviluppatori possono rinunciare a imparare a comporre Dockerfiles e concentrarsi sulla scrittura di codice. Man mano che le modifiche vengono unite nella base di codice, una pipeline CI/CD può compilare automaticamente la risorsa e quindi memorizzarla in un repository di artefatti. Infine, copiare l'artefatto nell'immagine appropriata prima di inviarlo a un registro Docker come Amazon ECR. Per lo meno dovresti creare un insieme di immagini di base da cui gli sviluppatori possono creare i propri

Dockerfiles. Evitare di estrarre immagini da Docker Hub. Non sempre sai cosa c'è nell'immagine e circa un quinto delle prime 1000 immagini presenta vulnerabilità. Un elenco di queste immagini e le loro vulnerabilità può essere trovato all'indirizzohttps://vulnerablecontainers.org/: .

Eseguire la scansione di pacchetti e librerie delle applicazioni per individuare eventuali vulnerabilità

L'uso di librerie open source è ormai comune. Come per i sistemi operativi e i pacchetti del sistema operativo, queste librerie possono avere vulnerabilità. Come parte del ciclo di vita di sviluppo, queste librerie devono essere analizzate e aggiornate quando vengono rilevate vulnerabilità critiche.

Docker Desktop esegue scansioni locali utilizzando Snyk. Può anche essere utilizzato per individuare vulnerabilità e potenziali problemi di licenza nelle librerie open source. Può essere integrato direttamente nei flussi di lavoro degli sviluppatori offrendo la possibilità di mitigare i rischi posti dalle librerie open source. Per ulteriori informazioni, consultare i seguenti argomenti:

- Strumenti di protezione delle applicazioni open sourceinclude un elenco di strumenti per rilevare le vulnerabilità nelle applicazioni.
- Scheda tecnica di scansione Docker

Eseguire l'analisi del codice statico

È necessario eseguire l'analisi del codice statico prima di creare un'immagine contenitore. Viene eseguito con il codice sorgente e viene utilizzato per identificare gli errori di codifica e il codice che potrebbero essere sfruttati da un attore malintenzionato, ad esempio le iniezioni di errore. SonarQubeè un'opzione popolare per i test statici di sicurezza delle applicazioni (SAST), con supporto per una varietà di diversi linguaggi di programmazione.

Esegui contenitori come utente non root

Dovresti eseguire i container come utente non root. Per impostazione predefinita, i contenitori vengono eseguiti come`root`utente a meno che il`USER`direttiva è inclusa nel Dockerfile. Le funzionalità Linux predefinite assegnate da Docker limitano le azioni che possono essere eseguite come`root`, ma solo marginalmente. Ad esempio, un contenitore in esecuzione come`root`non è ancora consentito accedere ai dispositivi.

Come parte della tua pipeline CI/CD, dovresti pelucchi Dockerfiles per cercare`USER`direttiva e fallire la build se manca. Per ulteriori informazioni, consultare i seguenti argomenti:

- DockerFile-Lintè uno strumento open source di RedHat che può essere utilizzato per verificare se il file è conforme alle best practice.
- Hadolintè un altro strumento per creare immagini Docker conformi alle best practice.

Utilizzare un file system radice di sola lettura

Dovresti usare un file system radice di sola lettura. Per impostazione predefinita, è possibile scrivere il file system principale di un container. Quando si configura un contenitore con unR0(sola lettura) file system root ti costringe a definire esplicitamente dove i dati possono essere persistenti. Ciò riduce la superficie di attacco perché il file system del contenitore non può essere scritto a meno che le autorizzazioni non siano specificamente concesse.

> **Note**
>
> Avere un file system root di sola lettura può causare problemi con alcuni pacchetti del sistema operativo che si aspettano di essere in grado di scrivere sul filesystem. Se si prevede di utilizzare file system radice di sola lettura, eseguire un test preliminare.

Configurare le attività con limiti di CPU e memoria (Amazon EC2)

È consigliabile configurare le attività con limiti di CPU e memoria per ridurre al minimo i seguenti rischi. I limiti delle risorse di un'attività impostano un limite superiore per la quantità di CPU e memoria che possono essere riservati da tutti i contenitori all'interno di un'attività. Se non sono impostati limiti, le attività hanno accesso alla CPU e alla memoria dell'host. Ciò può causare problemi in cui le attività distribuite su un host condiviso possono morire di fame altre attività delle risorse di sistema.

> **Note**
>
> Amazon ECS suAWS Fargaterichiedono di specificare i limiti della CPU e della memoria in quanto utilizza questi valori per scopi di fatturazione. Un'attività che monta tutte le risorse di sistema non è un problema per Amazon ECS Fargate perché ogni attività viene eseguita sulla propria istanza dedicata. Se non si specifica un limite di memoria, Amazon ECS assegna un minimo di 4 MB a ciascun contenitore. Analogamente, se non è impostato alcun limite di CPU per l'attività, l'agente contenitore Amazon ECS assegna un minimo di 2 CPU.

Utilizzare tag immutabili con Amazon ECR

Con Amazon ECR, puoi e dovresti usare le immagini di configurazione con tag immutabili. Ciò impedisce il push di una versione alterata o aggiornata di un'immagine nel repository di immagini con un tag identico. Questo protegge da un utente malintenzionato che spinge una versione compromessa di un'immagine sull'immagine con lo stesso tag. Usando tag immutabili, ti costringi effettivamente a spingere una nuova immagine con un tag diverso per ogni modifica.

Evitare l'esecuzione di container come privilegiato (Amazon EC2)

Dovresti evitare di eseguire i contenitori come privilegiati. Per lo sfondo, i contenitori vengono eseguiti come`privileged`vengono eseguiti con privilegi estesi sull'host. Ciò significa che il contenitore eredita tutte le funzionalità Linux assegnate a`root`sull'host. Il suo uso dovrebbe essere severamente limitato o proibito. Ti consigliamo di impostare la variabile d'ambiente dell'agente container di Amazon ECS`ECS_DISABLE_PRIVILEGED`Da a`true`per impedire l'esecuzione di contenitori come`privileged`su host particolari se`privileged`non è necessaria. In alternativa, puoi utilizzareAWS Lambdaper eseguire la scansione delle definizioni delle attività per l'utilizzo del`privileged`Parametro .

> **Note**
>
> Esecuzione di un contenitore come`privileged`Non è supportato da Amazon ECS suAWS Fargate: .

Rimuovere le funzionalità Linux non necessarie dal contenitore

Di seguito è riportato un elenco delle funzionalità Linux predefinite assegnate ai contenitori Docker. Per ulteriori informazioni su ciascuna funzionalità, consulta.Panoramica delle funzionalità Linux: .

```
CAP_CHOWN, CAP_DAC_OVERRIDE, CAP_FOWNER, CAP_FSETID, CAP_KILL,
CAP_SETGID, CAP_SETUID, CAP_SETPCAP, CAP_NET_BIND_SERVICE,
CAP_NET_RAW, CAP_SYS_CHROOT, CAP_MKNOD, CAP_AUDIT_WRITE,
CAP_SETFCAP
```

Se un contenitore non richiede tutte le funzionalità kernal di Docker elencate sopra, considera di eliminarle dal contenitore. Per ulteriori informazioni su ciascuna funzionalità kernal Docker,

consulta.KernalCapabilities: . Per ulteriori informazioni su quali funzionalità sono in uso, eseguendo le seguenti operazioni:

- Installare il pacchetto del sistema operativolibcap-nged eseguire`pscap`per elencare le funzionalità che ogni processo sta utilizzando.
- È possibile utilizzare ancheCapshper decifrare quali funzionalità sta usando un processo.
- Fare riferimento aFunzionalità Linux 101Per ulteriori informazioni.

Utilizzare una chiave cliente gestita (CMK) per crittografare le immagini inviate ad Amazon ECR

È consigliabile utilizzare una chiave gestita dal cliente (CMK) per incidere le immagini inviate ad Amazon ECR. Le immagini inviate ad Amazon ECR vengono crittografate automaticamente inattive con unAWS Key Management Service(AWS KMS) chiave gestita. Se preferisci usare la tua chiave, Amazon ECR ora supportaAWS KMScrittografia con chiavi gestite dal cliente (CMK). Prima di abilitare la crittografia lato server con un CMK, consultare le Considerazioni elencate nella documentazione suCrittografia inattiva: .

Sicurezza del runtime

La sicurezza runtime fornisce una protezione attiva per i container mentre sono in esecuzione. L'idea è quella di rilevare e prevenire attività dannose che si verifichino i tuoi contenitori.

Con elaborazione sicura (`seccomp`) è possibile impedire a un'applicazione containerizzata di effettuare determinati syscalls al kernel del sistema operativo host sottostante. Mentre il sistema operativo Linux ha poche centinaia di chiamate di sistema, la maggior parte di esse non sono necessarie per l'esecuzione di contenitori. Limitando quali syscall possono essere create da un contenitore, è possibile ridurre efficacemente la superficie di attacco dell'applicazione.

Per iniziare con seccomp, puoi usare`strace`per generare una traccia dello stack per vedere quali chiamate di sistema sta facendo l'applicazione. Puoi anche utilizzare uno strumento quale`syscall2seccomp`per creare un profilo seccomp dai dati raccolti dalla traccia dello stack. Per ulteriori informazioni, consultastraceesyscall2seccomp: .

A differenza del modulo di sicurezza SELinux, seccomp non può isolare i contenitori l'uno dall'altro. Tuttavia, protegge il kernel host da syscall non autorizzati. Funziona intercettando syscalls e permettendo solo a quelli che sono stati elencati di passare attraverso. Docker hapredefinitoprofilo seccomp adatto per la maggior parte dei carichi di lavoro generici.

> **Note**
>
> È inoltre possibile creare profili personalizzati per gli elementi che richiedono ulteriori privilegi.

AppArmor è un modulo di sicurezza Linux simile a seccomp, ma limita le capacità di un contenitore incluso l'accesso a parti del file system. Può essere eseguito in`enforcement`o`complain`Modalità. Dato che la creazione di profili AppArmor può essere difficile, consigliamo di utilizzare uno strumento comebane: . Per ulteriori informazioni su AppArmor, consulta laAppArmor(Certificato creato).

> **Important**
>
> AppArmor è disponibile solo per le distribuzioni Ubuntu e Debian di Linux.

Recommendations

Si consiglia di eseguire le azioni folowing durante la configurazione della sicurezza runtime.

Utilizzare una soluzione di terza parte per la difesa del runtime

Utilizzare una soluzione di terza parte per la difesa runtime. Se hai familiarità con il funzionamento della sicurezza Linux, crea e gestisci profili seccomp e AppArmor. Entrambi sono progetti open source. In caso contrario, è consigliabile utilizzare un servizio di terze parti diverso. La maggior parte utilizza l'apprendimento automatico per bloccare o segnalare attività sospette. Per un elenco delle soluzioni di terze parti disponibili, vedereMarketplace AWSPer Container: .

Aggiungere o rimuovere le funzionalità Linux utilizzando i criteri seccomp

Usa seccomp per avere un maggiore controllo sulle funzionalità Linux ed evitare errori di controllo syscall. Seccomp funziona come filtro syscall che revoca l'autorizzazione per eseguire determinati syscall o per utilizzare agruments specifici.

AWSPartner

Puoi anche utilizzare una qualsiasi delle seguentiAWSProdotti partner per aggiungere sicurezza e funzionalità aggiuntive ai tuoi carichi di lavoro Amazon ECS. Per ulteriori informazioni, consultaPartner Amazon ECS: .

Aqua Security

È possibile utilizzareAqua Securityper proteggere le applicazioni cloud-native dallo sviluppo alla produzione. Aqua Cloud Native Security Platform si integra con le risorse native del cloud e gli strumenti di orchestrazione per fornire sicurezza trasparente e automatizzata. Può prevenire attività sospette e attacchi in tempo reale e contribuire ad applicare le policy e semplificare la conformità alle normative.

Palo Alto Networks

Palo Alto Networksoffre sicurezza e protezione per host, container e infrastruttura serverless nel cloud e durante tutto il ciclo di vita di sviluppo e software.

Twistlock è fornito da Palo Alto Networks e può essere integrato con Amazon ECS FireLens. Con esso, si ha accesso a registri di sicurezza ad alta fedeltà e incidenti che sono facilmente aggregati in diversiAWSServizi . Questi includono Amazon CloudWatch, Amazon Athena e Amazon Kinesis. Twistlock protegge i carichi di lavoro distribuiti suAWSServizi di container.

Sysdig

È possibile utilizzareSysdigper eseguire carichi di lavoro nativi cloud sicuri e conformi negli scenari di produzione. Sysdig Secure DevOps Platform dispone di funzionalità integrate di sicurezza e conformità per proteggere i carichi di lavoro nativi per il cloud e offre scalabilità, prestazioni e personalizzazione di livello enterprise.

Cronologia dei documenti per la Guida per le best practice Amazon ECS

La tabella seguente descrive le versioni della documentazione per la Guida per le best practice Amazon ECS.

update-history-change	update-history-description	update-history-date
Best practice relative alla sicurezza	Sono state aggiunte le best practice per la gestione della sicurezza per i carichi di lavoro Amazon ECS.	26 maggio 2021
Procedure ottimali per la scalabilità automatica e la gestione della capacità	Sono state aggiunte le best practice per la scalabilità automatica e la gestione della capacità per i carichi di lavoro Amazon ECS.	14 maggio 2021
Best practice relative allo storage persistente	Sono state aggiunte le best practice per lo storage persistente per i carichi di lavoro Amazon ECS.	7 maggio 2021
Best practice relative alla rete	Sono state aggiunte le best practice per la gestione delle reti per i carichi di lavoro Amazon ECS.	6 Aprile 2021
Versione iniziale	Versione iniziale della Guida per le best practice Amazon ECS	6 Aprile 2021

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.