



Guida per gli sviluppatori

AWS Blockchain Templates



AWS Blockchain Templates: Guida per gli sviluppatori

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

I marchi e l'immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà delle rispettive aziende, che possono o meno essere associate, collegate o sponsorizzate da Amazon.

Table of Contents

.....	iv
Che cos'è AWS Blockchain Templates?	1
Come iniziare	2
Sono esperto con la blockchain AWS	2
Sono esperto AWS e nuovo nella blockchain	3
Sono un principiante AWS e competente con la blockchain	3
Sono nuovo nel mondo della blockchain AWS	3
Servizi correlati	3
Configurazione	5
Registrazione ad AWS	5
Creazione di un utente IAM	6
Crea una coppia di chiavi	8
Nozioni di base	9
Prerequisiti di installazione	10
Creazione di un VPC e delle sottoreti	10
Creazione di gruppi di sicurezza	14
Crea un ruolo IAM per Amazon ECS e un profilo di istanza EC2	16
Per creare un host bastione	22
Creazione della rete Ethereum	23
Connect EthStats e EthExplorer utilizzo di Bastion Host	26
Pulizia delle risorse	29
Modelli e caratteristiche di AWS Blockchain	31
Modello AWS Blockchain per Ethereum	31
Link per l'avvio	31
Opzioni Ethereum	32
Prerequisiti	35
Connessione a Ethereum Resources	43
Modello AWS Blockchain per Hyperledger Fabric	45
Link per l'avvio	45
Modello AWS Blockchain per componenti Hyperledger Fabric	46
Prerequisiti	47
Connessione alle risorse di Hyperledger Fabric	48
Cronologia dei documenti	50
Glossario AWS	51

AWSBlockchain Templates è stato interrotto il 30 aprile 2019. Non verranno effettuati ulteriori aggiornamenti a questo servizio o alla presente documentazione di supporto. Per la migliore esperienza di Managed Blockchain su AWS, ti consigliamo di utilizzare [Amazon Managed Blockchain \(AMB\)](#). Per ulteriori informazioni su come iniziare a usare Amazon Managed Blockchain, consulta il nostro [workshop su Hyperledger Fabric](#) o il nostro [blog sull'implementazione di un](#) nodo Ethereum. Se hai domande AMB o hai bisogno di ulteriore assistenza, contatta [AWS Supporto](#) il tuo AWS team addetto all'account.

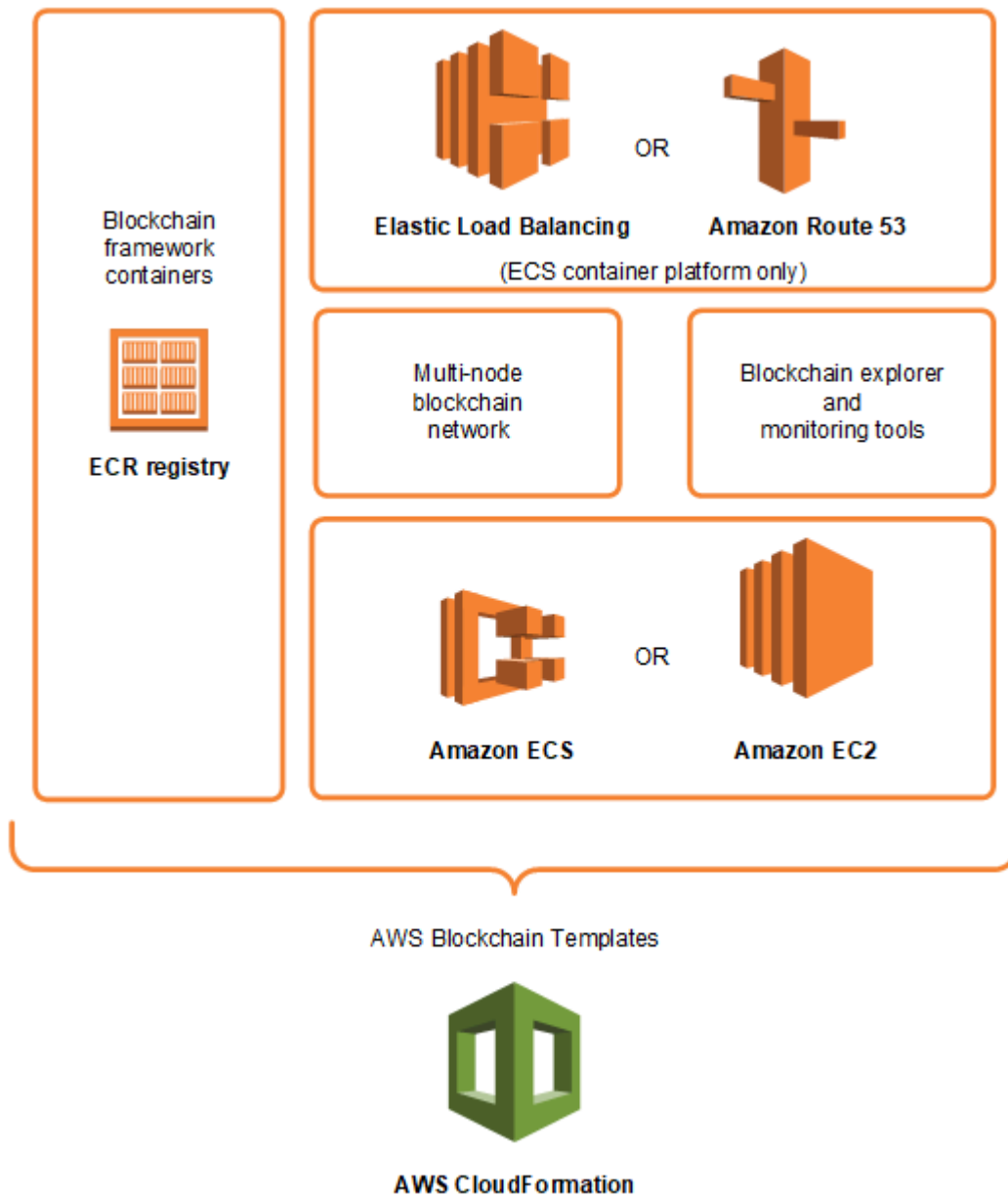
Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.

Che cos'è AWS Blockchain Templates?

AWS Blockchain Templates ti aiuta a creare e distribuire rapidamente reti blockchain AWS utilizzando diversi framework blockchain. Blockchain è una tecnologia di database decentralizzata che mantiene un set crescente di transazioni e contratti smart consolidati per ridurre il rischio di manomissioni, nonché revisioni utilizzando funzioni di crittografia.

Una rete blockchain è una peer-to-peer rete che migliora l'efficienza e l'immutabilità delle transazioni per processi aziendali come pagamenti internazionali, gestione della catena di approvvigionamento, registrazione catastale, crowdfunding, governance, transazioni finanziarie e altro ancora. In questo modo persone e organizzazioni che potrebbero non conoscersi si fidano le une delle altre e possono verificare in modo indipendente il record della transazione.

Utilizzi AWS Blockchain Templates per configurare e avviare AWS CloudFormation stack per creare reti blockchain. Le AWS risorse e i servizi che usi dipendono dal modello AWS Blockchain che scegli e dalle opzioni che specifichi. Per informazioni sui modelli disponibili e sulle relative funzionalità, consulta [Modelli e caratteristiche di AWS Blockchain](#). I componenti fondamentali di una rete blockchain AWS creata utilizzando AWS Blockchain Templates sono mostrati nel diagramma seguente.



Come iniziare

Il miglior punto di partenza dipende dal tuo livello di esperienza con la blockchain e, AWS in particolare, con i servizi relativi ad AWS Blockchain Templates.

Sono esperto con la blockchain AWS

Inizia consultando l'argomento in [Modelli e caratteristiche di AWS Blockchain](#) relativo al framework da utilizzare. Usa i link per avviare AWS Blockchain Template e configurare la rete blockchain oppure scarica i modelli per verificarli da solo.

Sono esperto AWS e nuovo nella blockchain

Inizia con il tutorial [Guida introduttiva a AWS Blockchain Templates](#), che ti guida attraverso il processo di creazione di una rete blockchain Ethereum introduttiva utilizzando le impostazioni predefinite. Al termine, per una panoramica dei framework blockchain e dei link e per ottenere ulteriori informazioni sulle opzioni di configurazione e sulle caratteristiche, consulta [Modelli e caratteristiche di AWS Blockchain](#).

Sono un principiante AWS e competente con la blockchain

Inizia consultando [Configurazione di modelli AWS Blockchain](#). Questo ti aiuta a configurare i fondamenti AWS, come un account e un profilo utente. Successivamente, esegui il tutorial [Guida introduttiva a AWS Blockchain Templates](#), che ti guida attraverso il processo di creazione di una rete blockchain Ethereum introduttiva. Anche se alla fine non utilizzerai Ethereum, avrai acquisito un'esperienza diretta con la configurazione dei servizi correlati. Questa esperienza è utile per tutti i framework blockchain. Infine, consulta l'argomento nella sezione [Modelli e caratteristiche di AWS Blockchain](#) relativa al tuo framework.

Sono nuovo nel mondo della blockchain AWS

Inizia consultando [Configurazione di modelli AWS Blockchain](#). Questo ti aiuta a configurare i fondamenti AWS, come un account e un profilo utente. Quindi, esegui il tutorial [Guida introduttiva a AWS Blockchain Templates](#), che ti guida attraverso il processo di creazione di una rete blockchain Ethereum introduttiva. Dedica del tempo a esplorare i link per saperne di più sui AWS servizi e su Ethereum.

Servizi correlati

A seconda delle opzioni selezionate, AWS Blockchain Templates può utilizzare i seguenti AWS servizi per distribuire la blockchain:

- Amazon EC2: fornisce capacità di calcolo per la tua rete blockchain. Per ulteriori informazioni, consulta la Guida per l'[utente di Amazon EC2](#).
- Amazon ECS: orchestra la distribuzione dei container tra le istanze EC2 in un cluster per la tua rete blockchain, se scegli di utilizzarla. Per ulteriori informazioni, consulta la [Guida per lo sviluppatore di Amazon Elastic Container](#).

- Amazon VPC: fornisce l'accesso alla rete per le risorse Ethereum che crei. Puoi personalizzare la configurazione per l'accessibilità e la sicurezza. Per ulteriori informazioni, consulta la [Amazon VPC Developer Guide](#).
- Bilanciamento del carico delle applicazioni: funge da unico punto di contatto per l'accesso alle interfacce utente disponibili e l'individuazione dei servizi interni quando si utilizza Amazon ECS come piattaforma container. Per ulteriori informazioni, consulta [Cos'è un Application Load Balancer?](#) nella Guida per l'utente di Application Load Balancers. .

Configurazione di modelli AWS Blockchain

Prima di iniziare con AWS Blockchain Templates, completa le seguenti attività:

- [Registrazione ad AWS](#)
- [Creazione di un utente IAM](#)
- [Crea una coppia di chiavi](#)

Questi sono i prerequisiti fondamentali per tutte le configurazioni della blockchain. Inoltre, la rete blockchain che scegli potrebbe avere dei prerequisiti che variano a seconda dell'ambiente e delle opzioni di configurazione desiderate. Per ulteriori informazioni, consulta la sezione pertinente per il tuo modello di blockchain in [Modelli e caratteristiche di AWS Blockchain](#).

Per step-by-step istruzioni su come configurare i prerequisiti per una rete Ethereum privata utilizzando un ECS cluster Amazon, consulta [Guida introduttiva a AWS Blockchain Templates](#)

Registrazione ad AWS

Quando ti iscrivi a AWS, il tuo AWS l'account viene registrato automaticamente per tutti i servizi. Ti vengono addebitati solo i servizi che utilizzi.

Se hai un AWS hai già un account, passa all'attività successiva. Se non hai un AWS account, usa la seguente procedura per crearne uno.

Per creare un AWS account

1. Apri la <https://portal.aws.amazon.com/billing/registrazione>.
2. Segui le istruzioni online.

Nel corso della procedura di registrazione riceverai una telefonata, durante la quale sarà necessario inserire un codice di verifica attraverso la tastiera del telefono.

Quando ti iscrivi a un Account AWS, un Utente root dell'account AWS viene creato. L'utente root ha accesso a tutti Servizi AWS e le risorse presenti nell'account. Come best practice di sicurezza, assegna l'accesso amministrativo a un utente e utilizza solo l'utente root per eseguire [attività che richiedono l'accesso di un utente root](#).

Nota il tuo AWS numero di conto. Ne hai bisogno quando crei un IAM utente nell'attività successiva.

Creazione di un utente IAM

Servizi in AWS richiedono che l'utente fornisca le credenziali al momento dell'accesso, in modo che il servizio possa determinare se l'utente dispone delle autorizzazioni per accedere alle sue risorse. Per la console è necessaria la password. È possibile creare chiavi di accesso per AWS account per accedere all'interfaccia della riga di comando oppure API. Tuttavia, ti sconsigliamo di accedere AWS utilizzando le credenziali per AWS account; ti consigliamo di utilizzare AWS Identity and Access Management (IAM) invece. Crea un IAM utente, quindi aggiungi l'utente a un IAM gruppo con autorizzazioni amministrative o concedi a questo utente le autorizzazioni amministrative. È quindi possibile accedere AWS utilizzando uno speciale URL e le credenziali per l'IAM utente.

Se ti sei registrato per AWS ma non hai ancora creato un IAM utente, puoi crearne uno utilizzando la IAM console. Se hai già un IAM utente, puoi saltare questo passaggio.

Per creare un utente amministratore, scegli una delle seguenti opzioni.

Scelta di un modo per gestire il tuo amministratore	Per	Come	Puoi anche
In IAM Identity Center (Consigliato)	Utilizza credenziali a breve termine per accedere AWS. Ciò è in linea con le best practice per la sicurezza. Per informazioni sulle migliori pratiche, consulta le migliori pratiche di sicurezza	Seguendo le istruzioni riportate in Guida introduttiva in AWS IAM Identity Center Guida per l'utente.	Configurare l'accesso programmatico configurando il AWS CLI da usare AWS IAM Identity Center nella AWS Command Line Interface Guida per l'utente.

Scelta di un modo per gestire il tuo amministratore	Per	Come	Puoi anche
	IAM nella Guida per l'IAMutente.		
In IAM (Non consigliato)	Usa credenziali a lungo termine per accedere AWS.	Segui le istruzioni riportate nella sezione Creazione del primo utente e gruppo di utenti IAM amministratore nella Guida per l'IAMutente.	Configura l'accesso programmatico gestendo le chiavi di accesso per IAM gli utenti nella Guida per l'IAMutente.

Per accedere come nuovo IAM utente, esci dal AWS Management Console, quindi usa quanto segue URL, dove `your_aws_account_id` è il tuo AWS numero di conto senza i trattini (ad esempio, se AWS il numero di conto è 1234-5678-9012, il tuo AWS l'ID dell'account è 123456789012):

```
https://your_aws_account_id.signin.aws.amazon.com/console/
```

Inserisci il nome IAM utente e la password che hai appena creato. Una volta effettuato l'accesso, la barra di navigazione visualizza "`your_user_name @ your_aws_account_id`".

Se non desideri che URL la pagina di accesso contenga i AWS ID account, puoi creare un alias per l'account. Dalla IAM dashboard, scegli Crea alias per l'account e inserisci un alias, ad esempio il nome della tua azienda. Per accedere dopo aver creato un alias per l'account, utilizza quanto segue: URL

```
https://your_account_alias.signin.aws.amazon.com/console/
```

Per verificare il link di accesso IAM degli utenti del tuo account, apri la IAM console e controlla il link di accesso IAM degli utenti nella dashboard.

Per ulteriori informazioni, consultare la [.AWS Guida per l'utente di Identity and Access Management.](#)

Crea una coppia di chiavi

AWS utilizza la crittografia a chiave pubblica per proteggere le informazioni di accesso per le istanze in una rete blockchain. Si specifica il nome della coppia di chiavi quando si utilizza ogni modello AWS Blockchain. È quindi possibile utilizzare la key pair per accedere direttamente alle istanze, ad esempio per accedere utilizzando SSH.

Se disponi già di una coppia di chiavi nella regione corretta, puoi ignorare questa fase. Se non hai già creato una key pair, puoi crearne una utilizzando la EC2 console Amazon. Crea una coppia di chiavi nella stessa regione utilizzata per avviare la rete Ethereum. Per ulteriori informazioni, consulta [Regioni e zone di disponibilità](#) nella Amazon EC2 User Guide.

Per creare una coppia di chiavi

1. Apri la EC2 console Amazon all'indirizzo <https://console.aws.amazon.com/ec2/>.
2. Dalla barra di navigazione, seleziona una regione per la coppia di chiavi. Puoi selezionare qualsiasi regione disponibile, indipendentemente dalla tua posizione, ma le coppie di chiavi sono specifiche di una regione. Ad esempio, se prevedi di avviare un'istanza nella regione Stati Uniti orientali (Ohio), devi creare una key pair per l'istanza nella stessa regione.
3. Nel riquadro di navigazione scegli Key Pairs (Coppie di chiavi), Create Key Pair (Crea coppia di chiavi).
4. In Key pair name (Nome coppia di chiavi), immetti un nome per la nuova coppia di chiavi. Scegli un nome facile da ricordare, ad esempio il tuo nome IAM utente, seguito da `-key-pair` e più il nome della regione. Ad esempio, `me-key-pair-useast2`. Scegli Create (Crea).
5. Il file della chiave privata viene automaticamente scaricato dal browser. Il nome di base del file è quello specificato come nome della coppia di chiavi e l'estensione è `.pem`. Salvare il file della chiave privata in un luogo sicuro.

Important

Questo è l'unico momento in cui salvare il file della chiave privata. Devi fornire il nome della tua coppia di chiavi quando avvii la rete Ethereum.

Per ulteriori informazioni, consulta [Amazon EC2 Key Pairs](#) nella Amazon EC2 User Guide. Per ulteriori informazioni sulla connessione alle EC2 istanze utilizzando la key pair, consulta [Connect to Your Linux Instance](#) nella Amazon EC2 User Guide.

Guida introduttiva a AWS Blockchain Templates

Questo tutorial dimostra come utilizzare AWS Blockchain Template per Ethereum per creare una rete blockchain privata su AWS Through. AWS CloudFormation La rete che crei ha due client Ethereum e un miner in esecuzione su istanze Amazon EC2 in un cluster Amazon ECS. Amazon ECS esegue questi servizi in contenitori Docker estratti da Amazon ECR. Prima di iniziare questo tutorial, è utile conoscere le reti blockchain e i AWS servizi coinvolti, ma non è obbligatorio.

Questo tutorial presuppone che tu abbia configurato i prerequisiti generali indicati in [Configurazione di modelli AWS Blockchain](#). Inoltre, devi configurare alcune AWS risorse, come una rete Amazon VPC e autorizzazioni specifiche per i ruoli IAM, prima di utilizzare il modello.

Il tutorial illustra come configurare tali prerequisiti. Sono state fatte delle scelte di configurazione, ma non sono prescrittive. Se si soddisfano i prerequisiti, è possibile effettuare altre scelte di configurazione in base alle esigenze dell'applicazione e dell'ambiente. Per informazioni sulle caratteristiche e i prerequisiti generali per ogni modello e per scaricare i modelli o lanciarli direttamente in AWS CloudFormation, consulta [Modelli e caratteristiche di AWS Blockchain](#).

In questo tutorial, gli esempi utilizzano la regione degli Stati Uniti occidentali (Oregon) (us-west-2), ma puoi utilizzare qualsiasi regione che supporti AWS Blockchain Templates:

- Regione Stati Uniti occidentali (Oregon) (us-west-2)
- Regione Stati Uniti orientali (Virginia settentrionale) (us-east-1)
- Regione Stati Uniti orientali (Ohio) (us-east-2)

Note

L'esecuzione di un modello in una regione non elencata sopra avvia risorse nella regione Stati Uniti orientali (Virginia settentrionale) (us-east-1).

Il modello AWS Blockchain per Ethereum che configuri utilizzando questo tutorial crea le seguenti risorse:

- Le istanze EC2 on demand del tipo e del numero da te specificati. Il tutorial utilizza il tipo di istanza t2.medium predefinito.

- Un Application Load Balancer interno.

Dopo il tutorial, viene illustrata la procedura per eliminare le risorse create.

Argomenti

- [Prerequisiti di installazione](#)
- [Creazione della rete Ethereum](#)
- [Connect EthStats e EthExplorer utilizzo di Bastion Host](#)
- [Pulizia delle risorse](#)

Prerequisiti di installazione

La configurazione di AWS Blockchain Template per Ethereum specificata in questo tutorial richiede quanto segue:

- [Creazione di un VPC e delle sottoreti](#)
- [Creazione di gruppi di sicurezza](#)
- [Crea un ruolo IAM per Amazon ECS e un profilo di istanza EC2](#)
- [Per creare un host bastione](#)

Creazione di un VPC e delle sottoreti

Il modello AWS Blockchain per Ethereum lancia le risorse in una rete virtuale definita dall'utente utilizzando Amazon Virtual Private Cloud (Amazon VPC). La configurazione specificata in questo tutorial crea un Application Load Balancer che richiede due sottoreti pubbliche in zone di disponibilità diverse. È inoltre necessaria una sottorete privata per le istanze di container e questa deve trovarsi nella stessa zona di disponibilità dell'Application Load Balancer. Utilizzare innanzitutto la procedura guidata del VPC per creare una sottorete pubblica e una sottorete privata nella stessa zona di disponibilità. Creare quindi una seconda sottorete pubblica all'interno di questo VPC in una zona di disponibilità diversa.

Per ulteriori informazioni, consultare [Che cos'è Amazon VPC?](#) nella Guida per l'utente di Amazon VPC

Utilizza la console Amazon VPC (<https://console.aws.amazon.com/vpc/>) per creare l'indirizzo IP elastico, il VPC e la sottorete come descritto di seguito.

Creazione di un indirizzo IP elastico

1. Apri alla console Amazon VPC all'indirizzo <https://console.aws.amazon.com/vpc/>.
2. Scegli Elastic IPs (IP elastici), Allocate new address (Alloca nuovo indirizzo), Allocate (Alloca).
3. Annota l'indirizzo IP elastico (EIP) creato, quindi seleziona Close (Chiudi).
4. Nell'elenco degli indirizzi IP elastici (EIP), trova Allocation ID (ID Allocazione) per l'indirizzo IP Elastico (EIP) creato in precedenza. Utilizzalo quando crei il VPC.

Per creare il VPC

1. Dalla barra di navigazione seleziona una regione per il VPC. I VPC sono specifici di una regione, quindi è necessario selezionare la stessa regione in cui è stata creata la coppia di chiavi e nella quale si sta avviando lo stack Ethereum. Per ulteriori informazioni, consulta [Crea una coppia di chiavi](#).
2. Nel pannello di controllo VPC scegli Start VPC Wizard (Avvia creazione guidata VPC).
3. Nella pagina Step 1: Select a VPC Configuration (Fase 1: Seleziona una configurazione VPC) scegliere VPC with Public and Private Subnets (VPC con sottoreti pubbliche e private), Select (Seleziona).
4. Nella pagina Step 2: VPC with Public and Private Subnets (Fase 2: VPC con sottoreti pubbliche e private) lascia i valori predefiniti per IPv4 CIDR block (Blocco CIDR IPv4) e per IPv6 CIDR block (Blocco CIDR IPv6). Per VPC name (Nome VPC) inserisci un nome descrittivo.
5. Per Public subnet's IPv4 CIDR (CIDR IPv4 della sottorete pubblica), lascia il valore predefinito. Per Availability Zone (Zona di disponibilità), scegli una zona. Per Public subnet name (Nome sottorete pubblica), inserisci un nome descrittivo.

Specifica questa sottorete come una delle prime due sottoreti per Application Load Balancer quando utilizzi il modello.

Prendi nota della zona di disponibilità di questa sottorete perché selezionerai la stessa zona di disponibilità per la sottorete privata e una diversa per l'altra sottorete pubblica.

6. Per Private subnet's IPv4 CIDR (CIDR IPv4 della sottorete privata), lascia il valore predefinito. Per Availability Zone (Zona di disponibilità), seleziona la stessa zona di disponibilità della fase precedente. Per Private subnet name (Nome sottorete privata), inserisci un nome descrittivo.
7. In Elastic IP Allocation ID (ID assegnazione IP elastico), seleziona l'indirizzo IP elastico (EIP) creato in precedenza.

8. Lascia le altre impostazioni ai valori predefiniti.
9. Seleziona Crea VPC.

L'esempio seguente mostra un VPC EthereumNetworkVPC con una sottorete pubblica EthereumPubSub1 e una sottorete privata 1. EthereumPvtSub La sottorete pubblica utilizza la zona di disponibilità us-west-2a.

Step 2: VPC with Public and Private Subnets

IPv4 CIDR block:* (65531 IP addresses available)

IPv6 CIDR block: No IPv6 CIDR Block
 Amazon provided IPv6 CIDR block

VPC name:

Public subnet's IPv4 CIDR:* (251 IP addresses available)

Availability Zone:* ▼

Public subnet name:

Private subnet's IPv4 CIDR:* (251 IP addresses available)

Availability Zone:* ▼

Private subnet name:

You can add more subnets after AWS creates the VPC.

Specify the details of your NAT gateway ([NAT gateway rates apply](#)). [Use a NAT instance instead](#)

Elastic IP Allocation ID:*

Service endpoints

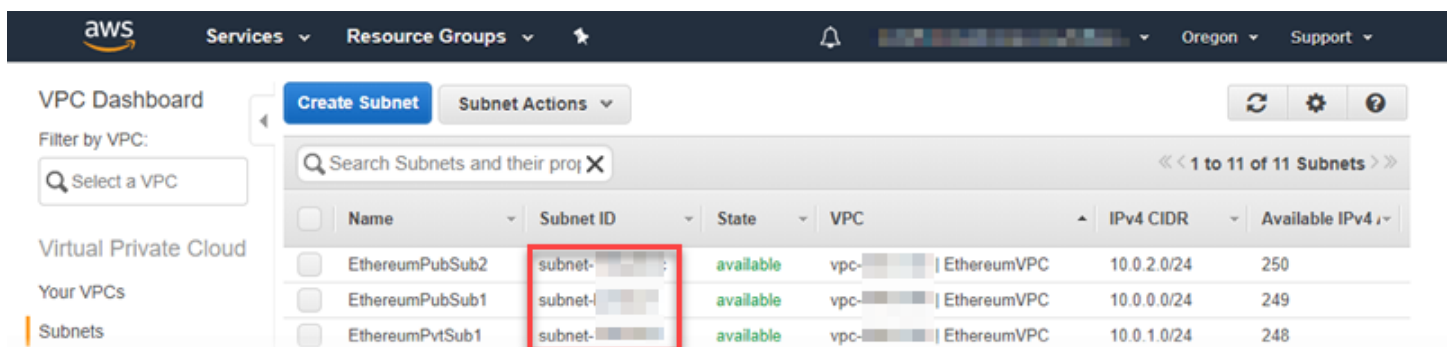
Enable DNS hostnames:* Yes No

Hardware tenancy:* ▼

Per creare la seconda sottorete pubblica in una diversa zona di disponibilità

1. Scegliere Subnets (Sottoreti) quindi selezionare dall'elenco la sottorete pubblica creata in precedenza. Selezionare la scheda Route Table (Tabella di routing) e prendere nota dell'ID di Route Table (Tabella di routing). Specificare la stessa tabella di routing per la seconda sottorete pubblica riportata di seguito.
2. Seleziona Create Subnet (Crea sottorete).
3. Per Name tag (Tag nome) immettere un nome per la sottorete. Utilizzare questo nome in seguito quando si crea il bastion host in questa rete.
4. In VPC, selezionare il VPC creato in precedenza.
5. Per Availability Zone (Zona di disponibilità), selezionare una zona diversa dalla quella selezionata per la prima sottorete pubblica.
6. Per IPv4 CIDR block (Blocco CIDR IPv4), immettere 10.0.2.0/24.
7. Selezionare Yes, Create (Sì, crea). La sottorete viene aggiunta all'elenco delle sottoreti.
8. Con la sottorete selezionata dall'elenco, scegliere Subnet Actions (Operazioni sottorete), Modify auto-assign IP settings (Modifica impostazioni di assegnazione automatica IP). Selezionare Auto-assign IPs (Assegnazione automatica IP), Save (Salva), Close (Chiudi). Ciò consente al bastion host di ottenere un indirizzo IP pubblico quando lo si crea in questa sottorete.
9. Nella scheda Route Table (Tabella di routing), scegliere Edit (Modifica). Per Change to (Cambia in), selezionare l'ID della tabella di routing annotata in precedenza e scegliere Save (Salva).

Ora dovresti vedere tre sottoreti per il VPC che hai creato in precedenza. Annota i nomi e gli ID delle tre sottoreti in modo che sia possibile specificarli utilizzando il modello.



Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4
EthereumPubSub2	subnet- [redacted]	available	vpc- [redacted] EthereumVPC	10.0.2.0/24	250
EthereumPubSub1	subnet- [redacted]	available	vpc- [redacted] EthereumVPC	10.0.0.0/24	249
EthereumPvtSub1	subnet- [redacted]	available	vpc- [redacted] EthereumVPC	10.0.1.0/24	248

Creazione di gruppi di sicurezza

I gruppi di sicurezza fungono da firewall, controllando il traffico in entrata e in uscita delle risorse. Quando utilizzi il modello per creare una rete Ethereum su un cluster Amazon ECS, specifichi due gruppi di sicurezza:

- Un gruppo di sicurezza per le istanze EC2 che controlla il traffico da e verso le istanze EC2 nel cluster
- Un gruppo di sicurezza per l'Application Load Balancer che controlla il traffico tra l'Application Load Balancer, le istanze EC2 e il bastion host. Questo gruppo di sicurezza viene associato anche al bastion host.

Ogni gruppo di sicurezza ha regole che consentono la comunicazione tra Application Load Balancer e le istanze EC2, nonché altre regole minime. Ciò richiede che i gruppi di sicurezza facciano riferimento l'uno all'altro. Per questo motivo, è necessario prima creare i gruppi di sicurezza e poi aggiornarli con le regole appropriate.

Per creare due gruppi di sicurezza

1. Apri la console Amazon EC2 all'indirizzo <https://console.aws.amazon.com/ec2/>.
2. Nel pannello di navigazione, scegli Gruppi di sicurezza, Crea gruppo di sicurezza.
3. Per Security group name (Nome gruppo di sicurezza), immetti un nome che sia semplice da identificare e che differenzi un gruppo dall'altro, ad esempio, EthereumEC2-SG o EthereumALB-SG. Utilizzerai questi nomi in seguito. Per Description (Descrizione) immetti un breve riepilogo.
4. In VPC, selezionare il VPC creato in precedenza.
5. Scegli Crea.
6. Ripeti le fasi precedenti per creare gli altri gruppi di sicurezza.


Aggiungi le regole in entrata al gruppo di sicurezza per le istanze EC2

1. Seleziona il gruppo di sicurezza per le istanze EC2 creato in precedenza
2. Nella scheda Inbound (In entrata), selezionare Edito (Modifica).
3. In Type (Tipo), selezionare All traffic (Tutto il traffico). Per Source (Origine), lascia selezionato Custom (Personalizzata), quindi scegli nell'elenco il gruppo di sicurezza che stai modificando, ad esempio, EthereumEC2-SG. In questo modo le istanze EC2 nel gruppo di sicurezza possono comunicare l'una con l'altra.

4. Selezionare Add Rule (Aggiungi regola).
5. In Type (Tipo), selezionare All traffic (Tutto il traffico). Per Source (Origine), lascia l'opzione Custom (Personalizzata) selezionata, quindi scegli nell'elenco il gruppo di sicurezza dell'Application Load Balancer, ad esempio, EthereumALB-SG. In questo modo le istanze EC2 nel gruppo di sicurezza possono comunicare con Application Load Balancer.
6. Selezionare Salva.

Aggiungi le regole in entrata e modifica le regole in uscita per il gruppo di sicurezza di Application Load Balancer

1. Seleziona il gruppo di sicurezza per i sistemi Application Load Balancer creato in precedenza
2. Nella scheda Inbound (In entrata) scegli Edit (Modifica), quindi aggiungi le seguenti regole in entrata:
 - a. In Type (Tipo), selezionare All traffic (Tutto il traffico). Per Source (Origine), lascia l'opzione Custom (Personalizzata) selezionata, quindi scegli nell'elenco il gruppo di sicurezza che stai modificando, ad esempio, EthereumALB-SG. Ciò consente all'Application Load Balancer di comunicare con se stesso e con il bastion host.
 - b. Selezionare Add Rule (Aggiungi regola).
 - c. In Type (Tipo), selezionare All traffic (Tutto il traffico). Per Source (Origine), lascia l'opzione Custom (Personalizzata) selezionata, quindi scegli nell'elenco il gruppo di sicurezza per le istanze EC2, ad esempio, EthereumEC2-SG. In questo modo le istanze EC2 nel gruppo di sicurezza possono comunicare con l'Application Load Balancer e con il bastion host.
 - d. Selezionare Add Rule (Aggiungi regola).
 - e. Per Type (Tipo) scegli SSH. In Source (Origine), selezionare My IP (IP personale), che rileva il CIDR IP del computer e lo inserisce.

 Important

Questa regola consente al bastion host di accettare il traffico SSH dal computer, consentendo ad esso di utilizzare il bastion host per visualizzare le interfacce Web e connettersi alle istanze EC2 sulla rete Ethereum. Per consentire ad altri utenti di collegarsi alla rete Ethereum, aggiungili come origini a questa regola. Consentire solo il traffico in entrata a origini attendibili.

- f. Selezionare Salva.

3. Nella scheda Outbound (In uscita), scegli Edit (Modifica) ed elimina la regola creata automaticamente per consentire il traffico in uscita su tutti gli indirizzi IP.
4. Selezionare Add Rule (Aggiungi regola).
5. In Type (Tipo), selezionare All traffic (Tutto il traffico). Per Destination (Destinazione), lascia l'opzione Custom (Personalizzata) selezionata e quindi scegli dall'elenco il gruppo di sicurezza per le istanze EC2. In questo modo vengono consentite le connessioni in uscita da Application Load Balancer e dal bastion host alle istanze EC2 nella rete Ethereum.
6. Selezionare Add Rule (Aggiungi regola).
7. In Type (Tipo), selezionare All traffic (Tutto il traffico). Per Destination (Destinazione), lascia l'opzione Custom (Personalizzata) selezionata, quindi scegli nell'elenco il gruppo di sicurezza che stai modificando, ad esempio, EthereumALB-SG. Ciò consente all'Application Load Balancer di comunicare con se stesso e con il bastion host.
8. Selezionare Salva.

Crea un ruolo IAM per Amazon ECS e un profilo di istanza EC2

Quando utilizzi questo modello, specifichi un ruolo IAM per Amazon ECS e un profilo di istanza EC2. Le policy di autorizzazione collegate a questi ruoli consentono alle risorse AWS e alle istanze presenti nel cluster di interagire con altre risorse AWS. Per ulteriori informazioni, consulta [Ruoli IAM](#) nella Guida per l'utente di IAM. Puoi configurare il ruolo IAM per Amazon ECS e il profilo dell'istanza EC2 utilizzando la console IAM (<https://console.aws.amazon.com/iam/>).

Per creare il ruolo IAM per Amazon ECS

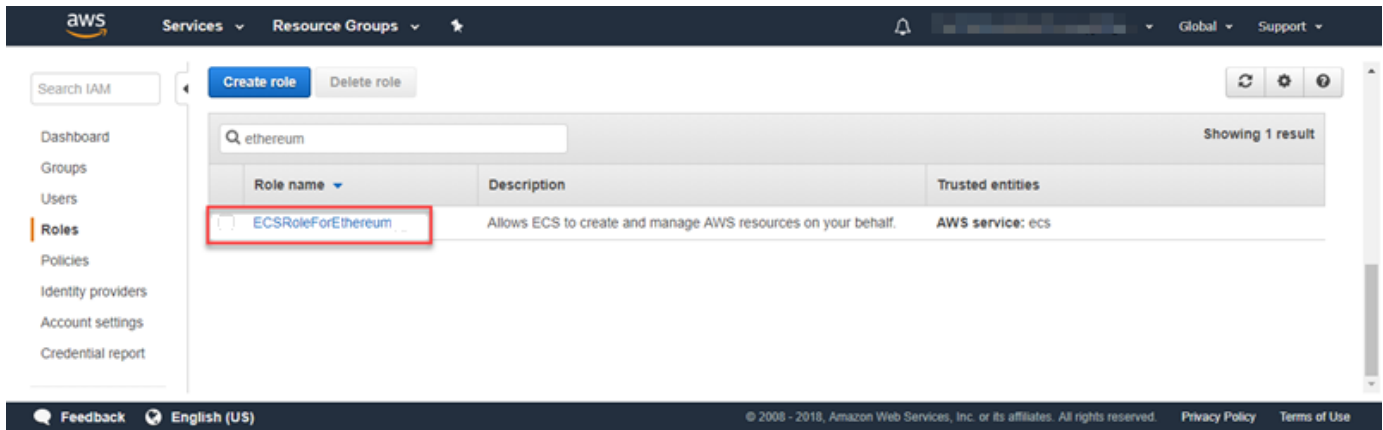
1. Aprire la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.
2. Nel riquadro di navigazione, seleziona Ruoli, quindi Crea nuovo ruolo.
3. In Select type of trusted entity (Seleziona tipo di entità attendibile), scegli AWS service (Servizio AWS).
4. In Choose the service that will use this role (Scegli il servizio che utilizzerà questo ruolo), selezionare Elastic Container Service.
5. In Select your use case (Seleziona il tuo caso d'uso), seleziona Elastic Container Service, Next:Permissions.

The screenshot shows the 'Create role' wizard in the AWS IAM console. It is divided into three steps:

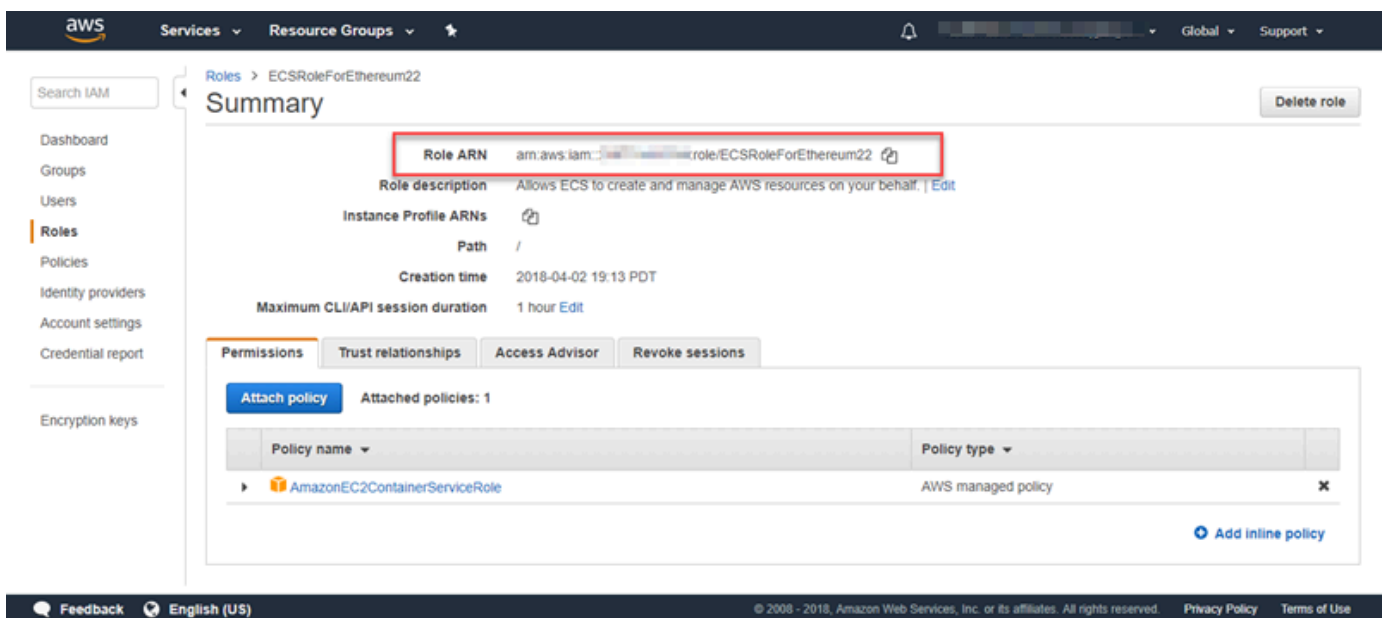
- Select type of trusted entity:** Four options are shown: 'AWS service' (selected), 'Another AWS account', 'Web identity', and 'SAML 2.0 federation'.
- Choose the service that will use this role:** A grid of AWS services is displayed. 'Elastic Container Service' is highlighted with a red box.
- Select your use case:** Two options are shown: 'EC2 Role for Elastic Container Service' and 'Elastic Container Service' (selected and highlighted with a red box).

At the bottom right, there are 'Cancel' and 'Next: Permissions' buttons. The footer contains 'Feedback', 'English (US)', and copyright information.

- Per la politica di autorizzazione, lascia selezionata la politica predefinita (Amazon EC2 ContainerServiceRole) e scegli Avanti:Revisione.
- Per Nome del ruolo, inserisci un valore che ti aiuti a identificare il ruolo, ad esempio ECS. RoleForEthereum Per Role Description (Descrizione ruolo) immetti un breve riepilogo. Annota il nome del ruolo per utilizzarlo in seguito.
- Scegli Crea ruolo.
- Dall'elenco, scegli il ruolo che hai appena creato. Se il tuo account ha molti ruoli, puoi cercare il nome del ruolo.



10. Copia il valore di Role ARN (ARN ruolo) e salvalo in modo che sia possibile copiarlo di nuovo. Hai bisogno di questo ARN quando crei la rete Ethereum.



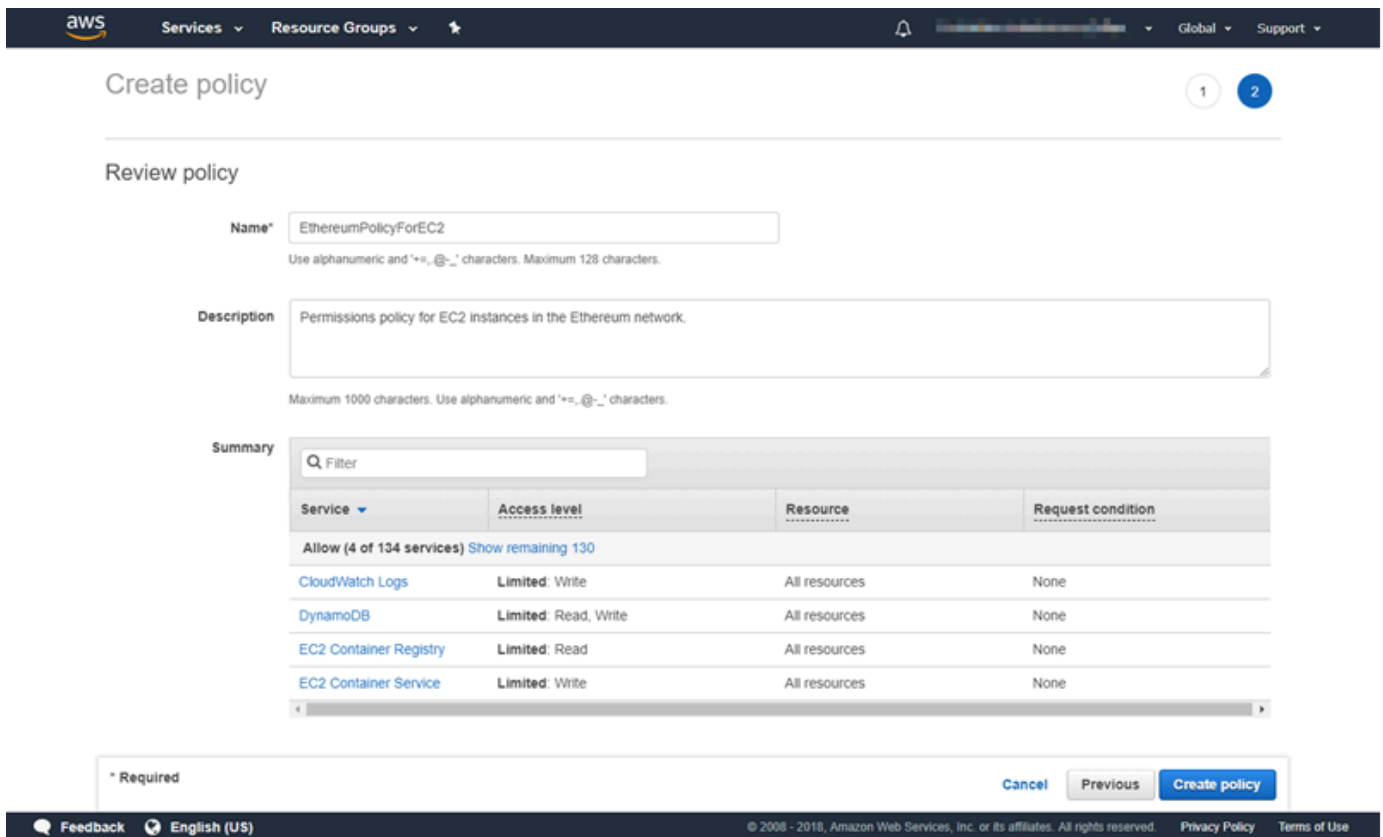
Il profilo dell'istanza EC2 specificato nel modello viene utilizzato dalle istanze EC2 nella rete Ethereum per interagire con altri servizi. AWS Puoi creare una policy di autorizzazione per il ruolo, creare il ruolo (che crea automaticamente un profilo dell'istanza con lo stesso nome) e quindi collegare la policy di autorizzazione al ruolo.

Per creare un profilo dell'istanza EC2

1. Nel riquadro di navigazione, seleziona Policy, quindi Crea policy.
2. Scegli JSON e sostituisci l'istruzione predefinita della policy con la seguente policy JSON:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ecs:CreateCluster",
        "ecs:DeregisterContainerInstance",
        "ecs:DiscoverPollEndpoint",
        "ecs:Poll",
        "ecs:RegisterContainerInstance",
        "ecs:StartTelemetrySession",
        "ecs:Submit*",
        "ecr:GetAuthorizationToken",
        "ecr:BatchCheckLayerAvailability",
        "ecr:GetDownloadUrlForLayer",
        "ecr:BatchGetImage",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "dynamodb:BatchGetItem",
        "dynamodb:BatchWriteItem",
        "dynamodb:PutItem",
        "dynamodb>DeleteItem",
        "dynamodb:GetItem",
        "dynamodb:Scan",
        "dynamodb:Query",
        "dynamodb:UpdateItem"
      ],
      "Resource": "*"
    }
  ]
}
```

3. Scegli Verifica policy.
4. Per Nome, inserisci un valore che ti aiuti a identificare questa politica di autorizzazioni, ad esempio EC2. EthereumPolicyFor Per Description (Descrizione) immetti un breve riepilogo. Scegli Crea policy.



Create policy 1 2

Review policy

Name* EthereumPolicyForEC2
Use alphanumeric and '+, @, _' characters. Maximum 128 characters.

Description Permissions policy for EC2 instances in the Ethereum network.
Maximum 1000 characters. Use alphanumeric and '+, @, _' characters.

Summary

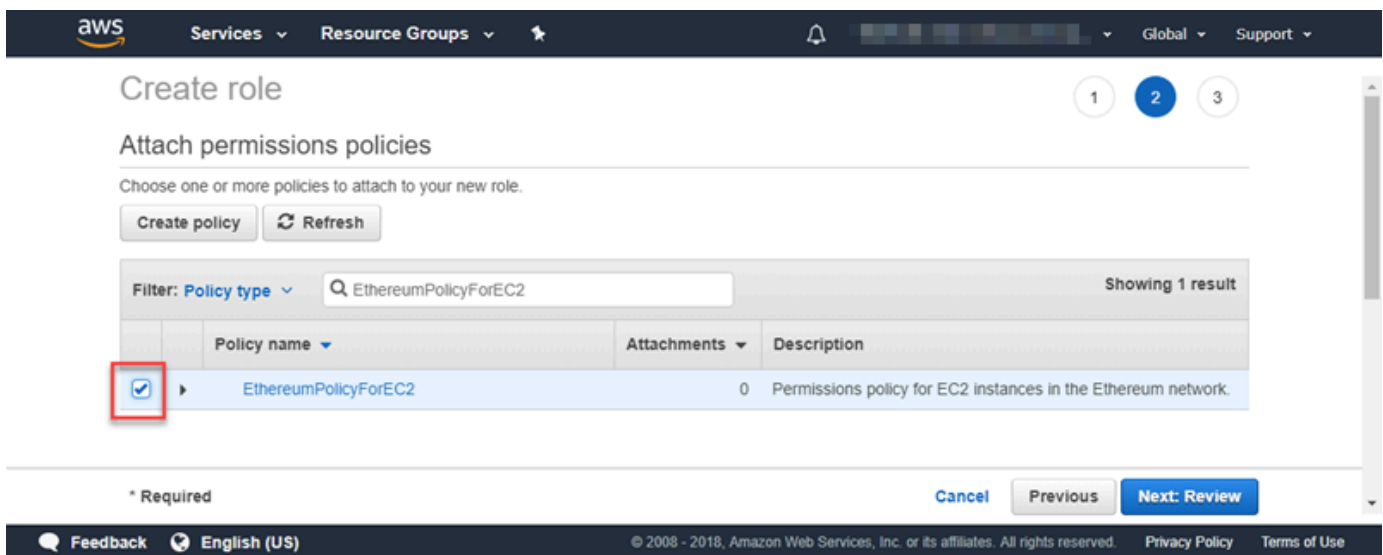
Filter

Service	Access level	Resource	Request condition
Allow (4 of 134 services) Show remaining 130			
CloudWatch Logs	Limited: Write	All resources	None
DynamoDB	Limited: Read, Write	All resources	None
EC2 Container Registry	Limited: Read	All resources	None
EC2 Container Service	Limited: Write	All resources	None

* Required Cancel Previous **Create policy**

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

5. Scegliere Roles (Ruoli), Create role (Crea ruolo).
6. Seleziona EC2, Next: Permissions (Successivo: autorizzazioni).
7. Nel campo Cerca, inserisci il nome della politica di autorizzazione che hai creato in precedenza, ad esempio EC2. EthereumPolicyFor
8. Seleziona il segno di spunta della policy creata in precedenza e scegli Next: Review (Successivo: Rivedi).



Create role 1 2 3

Attach permissions policies

Choose one or more policies to attach to your new role.

[Create policy](#) [Refresh](#)

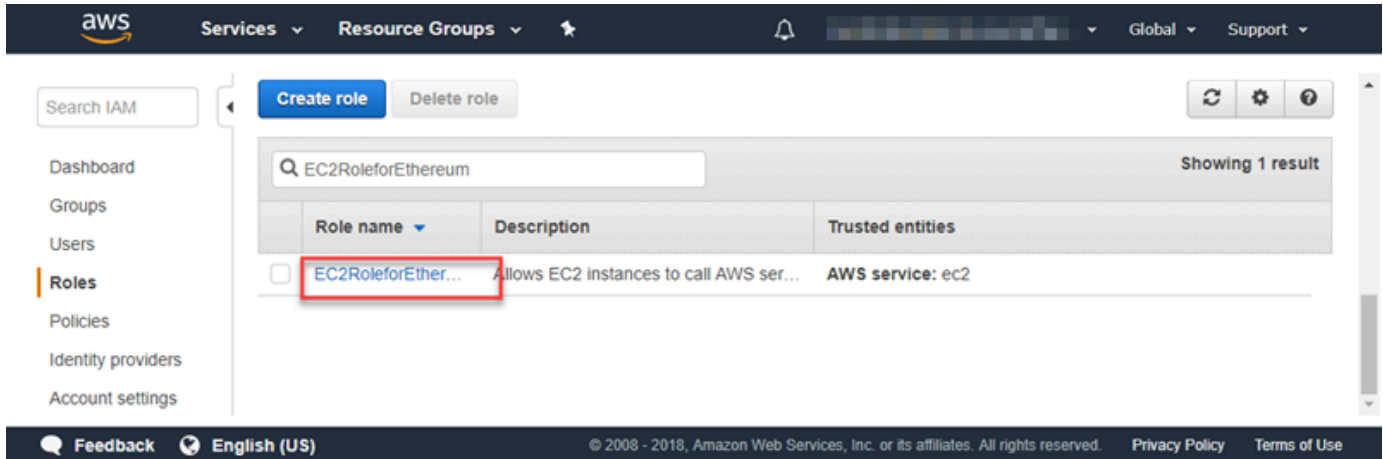
Filter: Policy type Showing 1 result

Policy name	Attachments	Description
<input checked="" type="checkbox"/> EthereumPolicyForEC2	0	Permissions policy for EC2 instances in the Ethereum network.

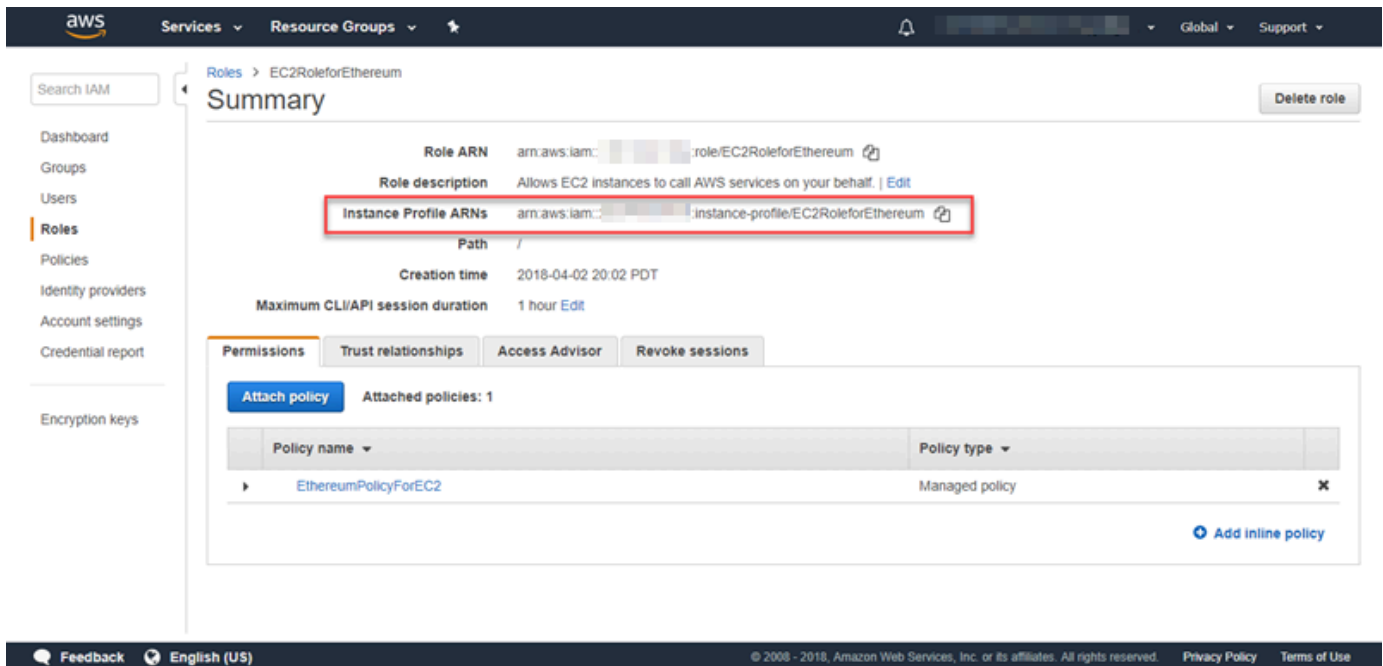
* Required Cancel Previous **Next: Review**

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

9. Per Nome del ruolo, inserisci un valore che ti aiuti a identificare il ruolo, ad esempio EC2. RoleForEthereum Per Role description (Descrizione ruolo), immetti un breve riepilogo. Scegli Create role (Crea ruolo).
10. Dall'elenco, scegli il ruolo che hai appena creato. Se il tuo account dispone di più ruoli, puoi inserire il nome del ruolo nel campo Search (Ricerca).



11. Copia il valore Instance Profile ARN (ARN profilo istanza) e salvalo in modo che sia possibile copiarlo di nuovo. Hai bisogno di questo ARN quando crei la rete Ethereum.



Per creare un host bastione

In questo tutorial viene creato un bastion host. Si tratta di un'istanza EC2 che usi per connetterti alle interfacce web e alle istanze della tua rete Ethereum. Il suo unico scopo è quello di inoltrare il traffico SSH da client attendibili al di fuori del VPC in modo che possano accedere alle risorse di rete Ethereum.

Si imposta il bastion host perché l'Application Load Balancer creato dal modello è interno, il che significa che instrada solo gli indirizzi IP interni. Il bastion host:

- Dispone di un indirizzo IP interno riconosciuto dall'Application Load Balancer perché viene lanciato nella seconda sottorete pubblica creata in precedenza.
- Dispone di un indirizzo IP pubblico assegnato dalla sottorete, a cui è possibile accedere da origini attendibili esterne al VPC.
- È associato al gruppo di sicurezza dell'Application Load Balancer creato in precedenza, che dispone di una regola in entrata che consente il traffico SSH (porta 22) dai client attendibili.

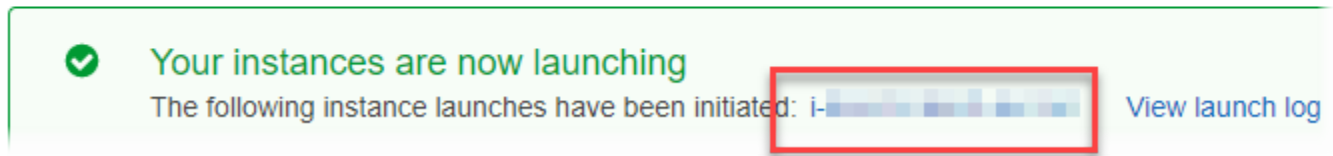
Per poter accedere alla rete Ethereum, è necessario configurare client attendibili per collegarsi mediante il bastion host. Per ulteriori informazioni, consulta [Connect EthStats e EthExplorer utilizzo di Bastion Host](#). Un bastion host è un approccio. È possibile utilizzare qualsiasi approccio che consenta di accedere da client attendibili a risorse private all'interno di un VPC.

Creazione di un bastion host

1. Segui i primi cinque passaggi per [lanciare un'istanza](#) nella Guida per l'utente di Amazon EC2.
2. Scegliere Edit Instance Details (Modifica dettagli istanza). Per Network (Rete), scegliere il VPC creato in precedenza, per Subnet (Sottorete) selezionare la seconda sottorete pubblica creata in precedenza. Lasciare tutte le altre impostazioni ai valori predefiniti.
3. Confermare la modifica quando richiesto, quindi scegliere Review and Launch (Verifica e avvia).
4. Scegliere Edit security groups (Modifica gruppi di sicurezza). Per Assign a security group (Assegna un gruppo di sicurezza), scegliere Select an existing security group (Seleziona un gruppo di sicurezza esistente).
5. Dall'elenco dei gruppi di sicurezza selezionare il gruppo di sicurezza dell'Application Load Balancer creato in precedenza, quindi scegliere Review and Launch (Verifica e avvia).
6. Scegli Avvia.

7. Prendere nota dell'ID istanza. Servirà più tardi, quando [Connect EthStats e EthExplorer utilizzo di Bastion Host](#).

Launch Status



Creazione della rete Ethereum

La rete Ethereum specificata utilizzando il modello in questo argomento avvia uno AWS CloudFormation stack che crea un cluster Amazon ECS di istanze EC2 per la rete Ethereum. Il modello si basa sulle risorse create in precedenza in [Prerequisiti di installazione](#).

Quando avvii lo AWS CloudFormation stack utilizzando il modello, crea stack annidati per alcune attività. Al completamento, è possibile connettersi alle risorse servite attraverso il sistema Application Load Balancer della rete attraverso il bastion host per verificare che la rete Ethereum sia in esecuzione e accessibile.

Per creare la rete Ethereum utilizzando il modello AWS Blockchain per Ethereum

1. Consulta la sezione [Getting Started with AWS Blockchain Templates](#) e apri la versione più recente di AWS Blockchain Template per Ethereum nella AWS CloudFormation console utilizzando i collegamenti rapidi per la tua regione AWS.
2. Inserisci i valori in base alle seguenti indicazioni:
 - Per Stack name (Nome stack) inserire un nome che sia facile da identificare. Il nome viene utilizzato all'interno dei nomi delle risorse create dallo stack.
 - In Ethereum Network Parameters (Parametri rete Ethereum) e Private Ethereum Network Parameters (Parametri rete privata Ethereum), lascia le impostazioni predefinite.

Warning

Utilizzare gli account predefiniti e la frase mnemonica associata solo a scopo di test. Non inviare Ether reale utilizzando il set predefinito di account perché chiunque abbia

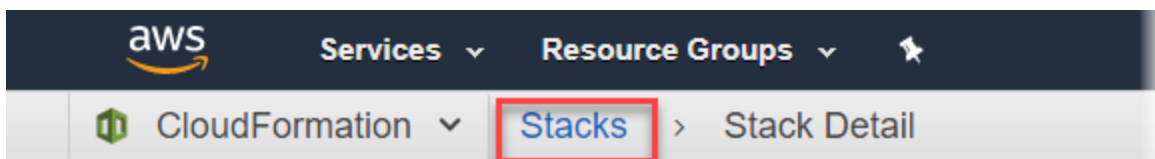
accesso alla frase mnemonica può accedere o rubare Ether dagli account. Specificare invece account personalizzati per scopi di produzione. La frase mnemonica associata all'account predefinito è `outdoor father modify clever trophy abandon vital feel portion grit evolve twist`.

- In Configurazione della piattaforma, lascia le impostazioni predefinite, che creano un cluster Amazon ECS di istanze EC2. L'alternativa, `docker-local`, crea una rete Ethereum mediante una singola istanza EC2.
- In EC2 configuration (Configurazione EC2), selezionare le opzioni in base alle seguenti linee guida:
 - In EC2 Key Pair (Coppia di chiavi EC2) seleziona una coppia di chiavi. Per informazioni su come creare una coppia di chiavi, consulta [Crea una coppia di chiavi](#).
 - Per EC2 Security Group (Gruppo di sicurezza EC2), seleziona il gruppo di sicurezza creato in precedenza in [Creazione di gruppi di sicurezza](#).
 - Per EC2 Instance Profile ARN (ARN profilo istanza EC2), immetti l'ARN del profilo dell'istanza creato in precedenza in [Crea un ruolo IAM per Amazon ECS e un profilo di istanza EC2](#).
- In VPC network configuration (Configurazione di rete VPC) selezionare le opzioni in base alle seguenti linee guida:
 - Per VPC ID (ID VPC) seleziona il VPC creato in precedenza in [Creazione di un VPC e delle sottoreti](#).
 - Per Ethereum Network Subnet IDs (ID sottorete di rete Ethereum), seleziona la singola sottorete privata creata in precedenza nella procedura [To create the VPC](#).
- In ECS cluster configuration (Configurazione cluster ECS), lasciare le impostazioni predefinite. In questo modo viene creato un cluster ECS di tre istanze EC2.
- In Application Load Balancer configuration (ECS only) (Configurazione Application Load Balancer (solo ECS)), selezionare le opzioni in base alle seguenti linee guida:
 - Per Application Load Balancer Subnet IDs (ID sottorete Application Load Balancer), selezionare le due sottoreti pubbliche da [list of subnets](#) annotate in precedenza.
 - Per Application Load Balancer Security Group (Gruppo di sicurezza di Application Load Balancer) seleziona il gruppo di sicurezza di Application Load Balancer creato in precedenza in [Creazione di gruppi di sicurezza](#).
 - Per IAM Role, inserisci l'ARN del ruolo ECS che hai creato in precedenza in [Crea un ruolo IAM per Amazon ECS e un profilo di istanza EC2](#)

- In EthStats, seleziona le opzioni in base alle seguenti linee guida:
 - Per Deploy EthStats, lascia l'impostazione predefinita, che è vera.
 - Per EthStats Connection Secret, digitate un valore arbitrario composto da almeno sei caratteri.
 - Sotto EthExplorer, lascia l'impostazione predefinita per Deploy EthExplorer, che è vera.
 - In Other parameters (Altri parametri), lascia il valore predefinito per Nested Template S3 URL Prefix Prefisso URL modello nidificato S3) e annotalo. Qui è dove puoi trovare i modelli nidificati.
3. Lascia tutte le altre impostazioni ai valori predefiniti, seleziona la casella di controllo di conferma, quindi scegli Create (Crea).

Viene visualizzata la pagina Stack Detail relativa allo stack principale che AWS CloudFormation viene avviato.

4. Per monitorare lo stato di avanzamento dello stack radice e degli stack nidificati scegli Stacks (Stack).



MyFirstEthereumStack

Stack name: MyFirstEthereumStack

5. Quando tutti gli stack mostrano CREATE_COMPLETE for Status, puoi connetterti alle interfacce utente di Ethereum per verificare che la rete sia attiva e accessibile. Quando si utilizza la piattaforma container ECS, gli URL per la connessione e l' EthJsonRPC tramite Application Load EthStats Balancer sono disponibili nella scheda Outputs dello stack principale. EthExplorer

Important

Non sarà possibile connettersi direttamente a questi URL o SSH fino a quando non si imposta una connessione proxy tramite bastion host sul computer client. Per ulteriori informazioni, consulta [Connect EthStats e EthExplorer utilizzo di Bastion Host](#).

The screenshot shows the AWS CloudFormation console. At the top, there are navigation menus for 'Services', 'Resource Groups', and 'Stacks'. Below this, there are buttons for 'Create Stack', 'Actions', and 'Design template'. A filter is set to 'Active' and 'By Stack Name'. A table lists four stacks, with the first one, 'MyFirstEthereumStack', selected. Below the table, there are tabs for 'Overview', 'Outputs', 'Resources', 'Events', 'Template', 'Parameters', 'Tags', 'Stack Policy', 'Change Sets', and 'Rollback Triggers'. The 'Outputs' tab is active, showing a table with three rows of output data. The first row, 'EthStatsURL', has its value highlighted with a red box.

Stack Name	Created Time	Status	Description
MyFirstEthereumStack-Ether... NESTED	2018-04-12 13:26:46 UTC-0700	CREATE_COMPLETE	This template creates an AutoScalingGroup of EC2 I...
MyFirstEthereumStack-Ether... NESTED	2018-04-12 13:26:38 UTC-0700	CREATE_COMPLETE	This template creates the ECS cluster and Ethereu...
MyFirstEthereumStack-Ether... NESTED	2018-04-12 13:25:59 UTC-0700	CREATE_COMPLETE	This template deploys an Ethereum cluster on an ex...
<input checked="" type="checkbox"/> MyFirstEthereumStack	2018-04-12 13:25:54 UTC-0700	CREATE_COMPLETE	This template creates an Ethereum network on an A...

Key	Value	Description	Export Name
EthStatsURL	http://MyFir-... ...us-west-2.elb.amazonaws.com	Visit this URL to see the status of your ...	
EthExplorerURL	http://MyFir-... ...us-west-2.elb.amazonaws.com:80 80	Visit this URL to view transactions on yo...	
EthJsonRPCURL	http://MyFir-... ...us-west-2.elb.amazonaws.com:85 45	Use this URL to access the Geth JSON ...	

Connect EthStats e EthExplorer utilizzo di Bastion Host

Per connettersi alle risorse Ethereum in questo tutorial, è possibile configurare l'inoltro delle porte SSH (tunneling SSH) attraverso il bastion host. Le seguenti istruzioni mostrano come eseguire questa operazione in modo da poterti connettere EthStats agli EthExplorer URL utilizzando un browser. Nelle istruzioni riportate di seguito, è necessario innanzitutto impostare un proxy SOCKS su una porta locale. Quindi utilizzi un'estensione del browser per utilizzare questa porta di inoltro per gli URL della tua rete Ethereum. [FoxyProxy](#)

Se usi Mac OS o Linux, usa un client SSH per configurare la connessione proxy SOCKS al bastion host. Se sei un utente Windows, usa PuTTY. Prima di connettersi, verificare che il computer client in uso sia specificato come origine consentita per il traffico SSH in entrata nel gruppo di sicurezza dell'Application Load Balancer configurato in precedenza.

Connessione al bastion host con l'inoltro alla porta SSH mediante SSH

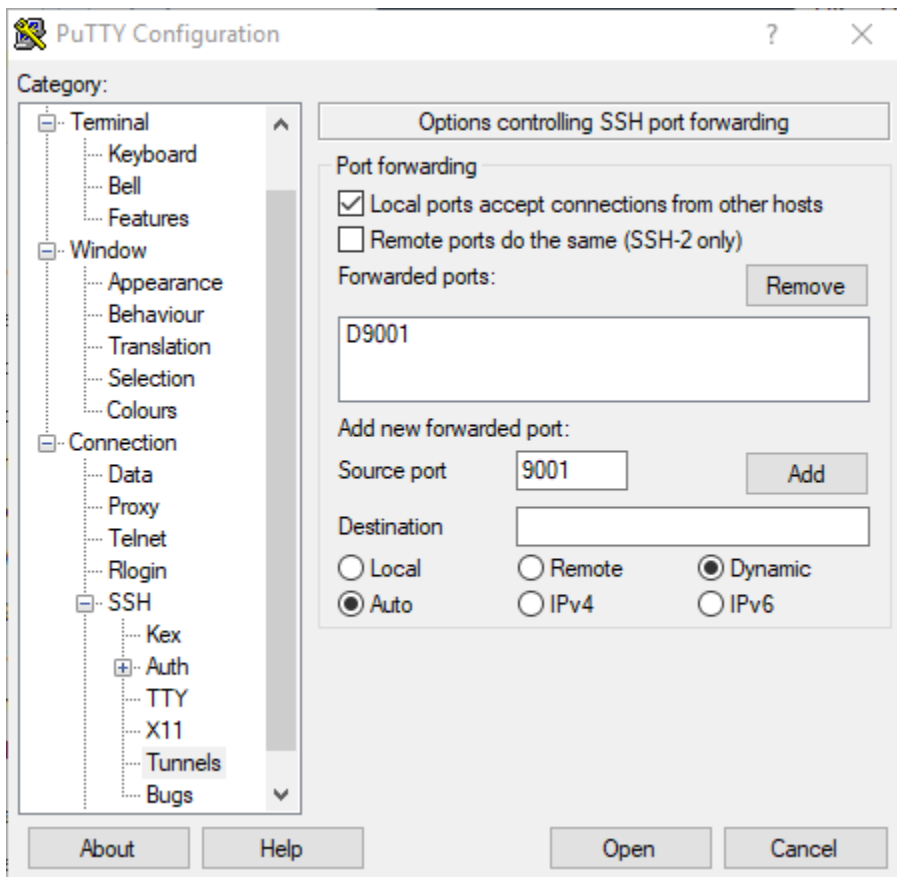
- Segui le procedure in [Connessione alla tua istanza Linux tramite SSH nella Guida](#) per l'utente di Amazon EC2. Per il passaggio 4 della procedura [Connecting to Your Linux Instance](#), aggiungi -D 9001 il comando SSH, specifica la stessa coppia di chiavi specificata nella configurazione di AWS Blockchain Template for Ethereum e specifica il nome DNS dell'host bastion.

```
ssh -i /path/my-template-key-pair.pem ec2-user@bastion-host-dns -D 9001
```

Connessione al bastion host con l'inoltro alla porta SSH mediante PuTTY (Windows)

1. Segui le procedure riportate in [Connessione alla tua istanza Linux da Windows con PuTTY](#) nella Guida per l'utente di Amazon EC2 fino al passaggio 7 della procedura [Avvio di una sessione PuTTY, utilizzando la stessa coppia di chiavi specificata nella configurazione di AWS Blockchain Template per Ethereum](#).
2. In PuTTY, in Category (Categoria), scegliere Connection (Connessione), SSH, Tunnels (Tunnel).
3. Per Port forwarding (Inoltro alla porta), scegliere Local ports accept connections from other hosts (Le porte locali accettano connessioni da altri host).
4. In Add new forwarded port (Aggiungi nuova porta inoltrata):
 - a. Per Source port (Porta di origine), immettere 9001. Questa è una porta arbitraria inutilizzata che abbiamo scelto; è possibile scegliere una porta diversa, se necessario.
 - b. Lasciare vuoto Destination (Destinazione).
 - c. Selezionare Dynamic (Dinamico).
 - d. Scegli Aggiungi.

Per Forwarded ports (Porte inoltrate), D9001 dovrebbe apparire come illustrato di seguito.



5. Scegliere Open (Apri) e quindi eseguire l'autenticazione per il bastion host come richiesto dalla configurazione della chiave. Lasciare aperta la connessione.

Con la connessione PuTTY aperta, è ora possibile configurare il sistema o un'estensione del browser per utilizzare la porta inoltrata per gli URL di rete Ethereum. Le seguenti istruzioni si basano sull'utilizzo di FoxyProxy Standard per inoltrare le connessioni in base al pattern URL di EthStats EthExplorer and e alla porta 9001, che hai stabilito in precedenza come porta di inoltro, ma puoi utilizzare qualsiasi metodo che preferisci.

Per configurare l'utilizzo del tunnel SSH FoxyProxy per gli URL della rete Ethereum

Questa procedura è stata scritta sulla base di Chrome. Se utilizzi un altro browser, traduci le impostazioni e la sequenza nella versione di FoxyProxy quel browser.

1. Scaricate e installate l'estensione FoxyProxy Standard del browser, quindi aprite Opzioni in base alle istruzioni del browser.
2. Selezionare Add new profile (Aggiungi nuovo profilo).

3. Nella scheda General (Impostazioni generali) assicurarsi che il proxy sia Enabled (Abilitato) e immettere Proxy Name (Nome proxy) e Proxy Notes (Note proxy) che consentono di identificare questa configurazione proxy.
4. Nella scheda Proxy Details (Dettagli proxy) scegliere Manual Proxy Configuration (Configurazione manuale proxy). Per Host or IP Address (Host o Indirizzo IP) (oppure Server or IP Address (Server o Indirizzo IP) in alcune versioni), immettere localhost. Per Port (Porta), immettere 9001. Selezionare SOCKS proxy? (Proxy SOCKS?).
5. Nella scheda URL Pattern (Modello URL) scegliere Add New Pattern (Aggiungi nuovo modello).
6. Per Pattern name, inserisci un nome facile da identificare e per URL Pattern, inserisci un pattern che corrisponda a tutti gli URL delle risorse Ethereum che hai creato con il modello, ad esempio `http://internal - MyUser -loadB-*`. Per informazioni sulla visualizzazione degli URL, consulta [Ethereum URLs](#).
7. Lasciare le selezioni predefinite per le altre impostazioni e scegliere Save (Salva).

Ora puoi connetterti agli URL di Ethereum, disponibili sulla CloudFormation console utilizzando la scheda Outputs dello stack principale che hai creato con il modello.

Pulizia delle risorse

AWS CloudFormation semplifica la pulizia delle risorse create dallo stack. Quando si elimina lo stack, tutte le risorse create dallo stack vengono eliminate.

Per eliminare le risorse create dal modello

- Apri la AWS CloudFormation console, seleziona lo stack principale che hai creato in precedenza, scegli Azioni, Elimina.

Lo Status (Stato) dello stack radice creato in precedenza e l'aggiornamento degli stack nidificati associati assumono lo stato `DELETE_IN_PROGRESS`.

Puoi scegliere di eliminare i prerequisiti creati per la rete Ethereum.

Eliminare il VPC

- Apri la console Amazon VPC, seleziona il VPC che hai creato in precedenza, quindi scegli Azioni, Elimina VPC. Questa operazione elimina anche le sottoreti, i gruppi di sicurezza e il gateway NAT associati al VPC.

Eliminare il ruolo IAM e il profilo dell'istanza EC2

- Apri la console IAM e scegli Ruoli. Seleziona il ruolo per ECS e il ruolo per EC2 precedentemente creati e scegli Delete (Elimina).

Terminare l'istanza EC2 per il bastion host

- Apri la dashboard di Amazon EC2, scegli Running instances, seleziona l'istanza EC2 che hai creato per l'host bastion, scegli Actions, Instance State, Terminate.

Modelli e caratteristiche di AWS Blockchain

In questa sezione vengono forniti i link necessari per iniziare a creare subito la rete blockchain, nonché informazioni sulle opzioni di configurazione e i prerequisiti per la configurazione della rete su AWS.

Sono disponibili i seguenti modelli:

- [Modello AWS Blockchain per Ethereum](#)
- [Modello AWS Blockchain per Hyperledger Fabric](#)

AWS Blockchain Templates è disponibile nelle seguenti regioni:

- Regione Stati Uniti occidentali (Oregon) (us-west-2)
- Regione Stati Uniti orientali (Virginia settentrionale) (us-east-1)
- Regione Stati Uniti orientali (Ohio) (us-east-2)

Note

L'esecuzione di un modello in una regione non elencata sopra avvia risorse nella regione Stati Uniti orientali (Virginia settentrionale) (us-east-1).

Utilizzo del modello AWS Blockchain per Ethereum

Ethereum è un framework blockchain che esegue contratti smart tramite Solidity, un linguaggio specifico di Ethereum. Homestead è la versione più recente di Ethereum. [Per ulteriori informazioni, consulta la documentazione di Ethereum Homestead e la documentazione di Solidity.](#)

Link per l'avvio

Consulta [Getting Started with AWS Blockchain Templates](#) per i link da avviare AWS CloudFormation in regioni specifiche utilizzando i modelli Ethereum.

Opzioni Ethereum

Quando configuri la rete Ethereum utilizzando il modello, puoi effettuare delle scelte che determinano i requisiti successivi:

- [Scelta della piattaforma dei container](#)
- [Scelta di una rete Ethereum privata o pubblica](#)
- [Modifica degli account predefiniti e della frase mnemonica](#)

Scelta della piattaforma dei container

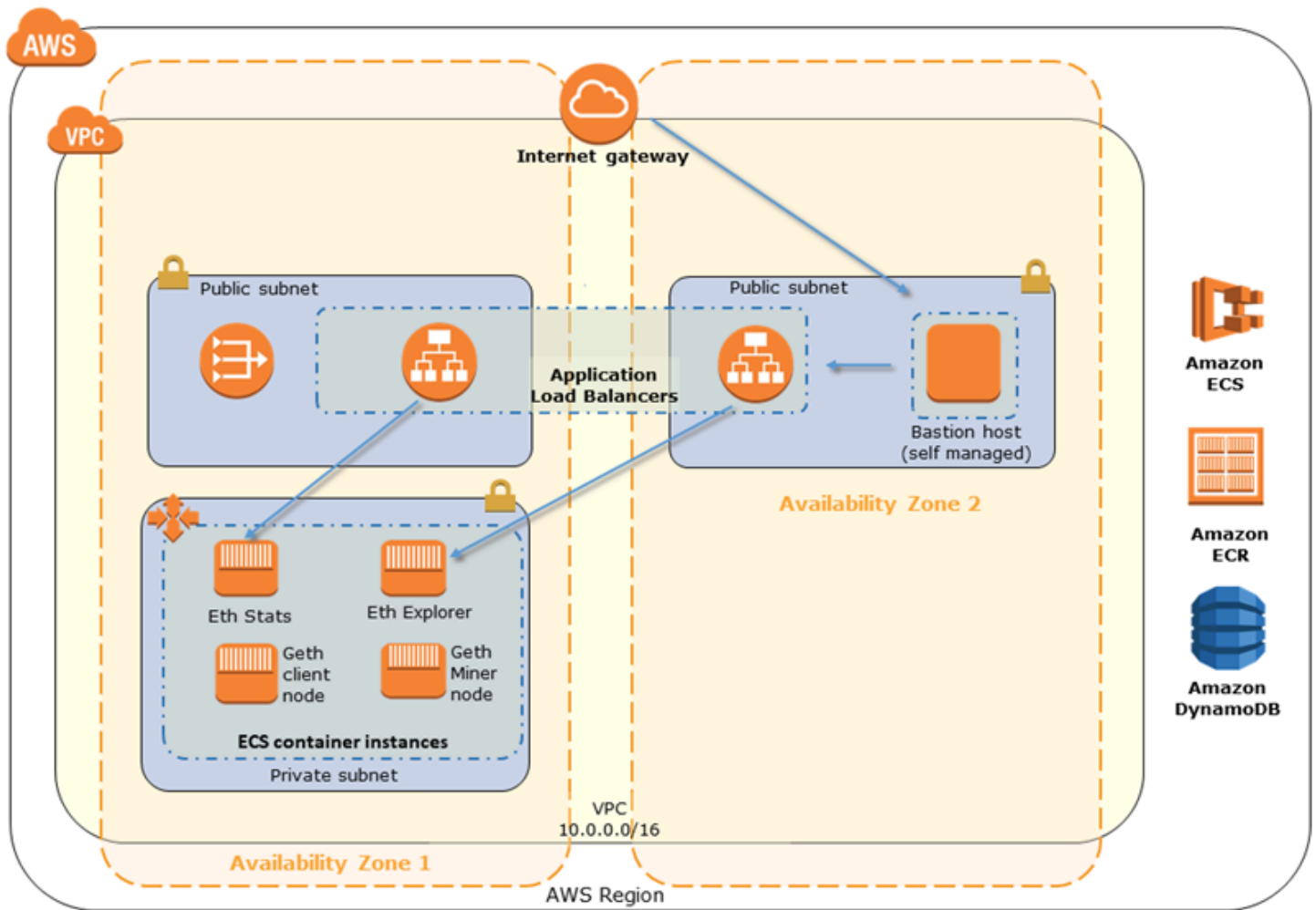
AWS Blockchain Templates utilizza contenitori Docker archiviati in Amazon ECR per distribuire software blockchain. Il modello AWS Blockchain per Ethereum offre due scelte per la piattaforma container:

- `ecs`: specifica che Ethereum viene eseguito su un cluster Amazon ECS di istanze Amazon EC2.
- `docker-local`: specifica che Ethereum viene eseguito su una singola istanza EC2.

Utilizzo della piattaforma container Amazon ECS

Con Amazon ECS, crei la tua rete Ethereum su un cluster ECS composto da più istanze EC2, con un Application Load Balancer e risorse correlate. Per ulteriori informazioni sull'uso della configurazione di Amazon ECS, consulta il [Guida introduttiva a AWS Blockchain Templates](#) tutorial.

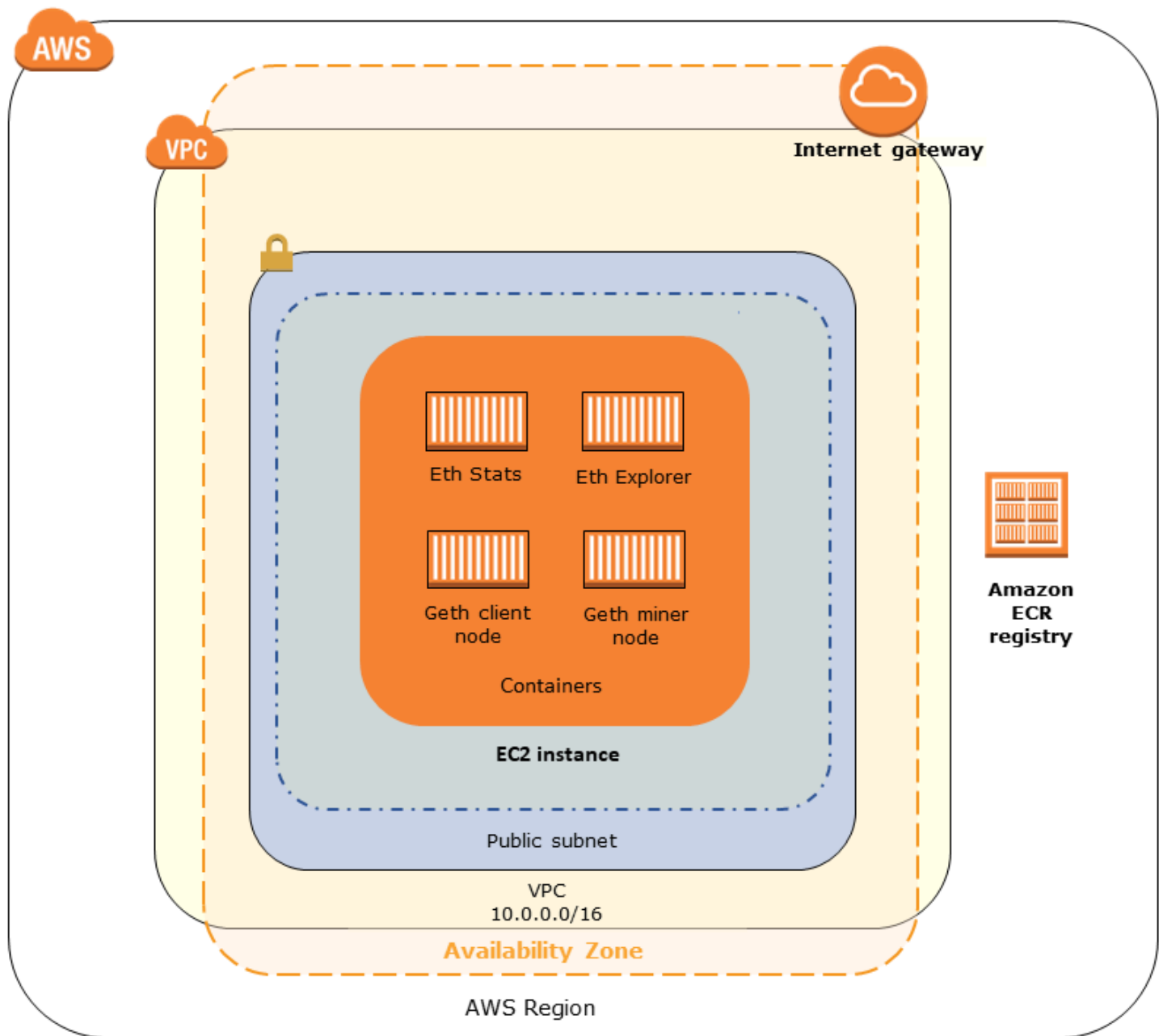
Il diagramma seguente mostra una rete Ethereum creata utilizzando il modello con l'opzione della piattaforma container ECS:



Utilizzo della piattaforma docker-local

In alternativa, puoi avviare contenitori Ethereum all'interno di una singola istanza Amazon EC2. Tutti i container vengono eseguiti su una singola istanza EC2. Questa è una configurazione semplificata.

Il diagramma seguente mostra una rete Ethereum creata utilizzando il modello con l'opzione docker-local container platform:



Scelta di una rete Ethereum privata o pubblica

La scelta di un valore per l'ID di rete Ethereum diverso da 1-4 crea nodi Ethereum privati che vengono eseguiti all'interno di una rete che hai definito, utilizzando i parametri della rete privata da te specificati.

Quando scegli un ID di rete Ethereum compreso tra 1 e 4, i nodi Ethereum che crei vengono uniti alla rete pubblica Ethereum. Puoi ignorare le impostazioni della rete privata e le relative impostazioni

predefinite. Se scegli di unire i nodi Ethereum alla rete Ethereum pubblica, assicurati che i servizi appropriati nella tua rete siano accessibili da Internet.

Modifica degli account predefiniti e della frase mnemonica

Una frase mnemonica è un insieme casuale di parole che è possibile utilizzare per generare wallet Ethereum (cioè coppie di chiavi private/pubbliche) per gli account associati su qualsiasi rete. La frase mnemonica può essere utilizzata per l'accesso a Ether da parte degli account associati. Abbiamo creato un mnemonico predefinito associato agli account predefiniti utilizzati dal modello Ethereum.

Warning

Utilizzare gli account predefiniti e la frase mnemonica associata solo a scopo di test. Non inviare Ether reale utilizzando il set predefinito di account perché chiunque abbia accesso alla frase mnemonica può accedere o rubare Ether dagli account. Specificare invece account personalizzati per scopi di produzione. La frase mnemonica associata all'account predefinito è `outdoor father modify clever trophy abandon vital feel portion grit evolve twist`.

Prerequisiti

Quando configuri la tua rete Ethereum utilizzando il modello AWS Blockchain per Ethereum, devono essere soddisfatti i requisiti minimi elencati di seguito. Il modello richiede i componenti AWS elencati per ciascuna delle seguenti categorie:

Argomenti

- [Prerequisiti per l'accesso alle risorse di Ethereum](#)
- [Prerequisiti IAM](#)
- [Prerequisiti per i gruppi di sicurezza](#)
- [Prerequisiti per VPC](#)
- [Esempi di autorizzazioni IAM per il profilo di istanza EC2 e il ruolo ECS](#)

Prerequisiti per l'accesso alle risorse di Ethereum

Prerequisito	Per la piattaforma ECS	Per docker-local
<p>Una coppia di chiavi Amazon EC2 che puoi usare per accedere alle istanze EC2. La chiave deve esistere nella stessa regione del cluster ECS e delle altre risorse.</p>	✓	✓
<p>Un componente connesso a Internet, ad esempio un bastion host o un sistema di bilanciamento del carico connesso a Internet, con un indirizzo interno dal quale è consentito il traffico verso l'Application Load Balancer. Questa operazione è necessaria per la piattaforma ECS perché il modello crea un sistema di bilanciamento del carico interno per motivi di sicurezza. Questa operazione è necessaria con la piattaforma docker-local quando l'istanza EC2 si trova in una sottorete privata, che è l'opzione consigliata. Per informazioni sulla configurazione di un bastion host, consulta Per creare un host bastione.</p>	✓	✓ (con sottorete privata)

Prerequisiti IAM

Prerequisito	Per la piattaforma ECS	Per docker-local
Un responsabile IAM (utente o gruppo) che dispone delle autorizzazioni per lavorare con tutti i servizi correlati.	✓	✓
Un profilo di istanza Amazon EC2 con autorizzazioni appropriate per consentire alle istanze EC2 di interagire con altri servizi. Per ulteriori informazioni, consulta To create an EC2 instance profile.	✓	✓
Un ruolo IAM con autorizzazioni per Amazon ECS per interagire con altri servizi. Per ulteriori informazioni, consulta Creazione del ruolo ECS e delle autorizzazioni.	✓	

Prerequisiti per i gruppi di sicurezza

Prerequisito	Per la piattaforma ECS	Per docker-local
Un gruppo di sicurezza per le istanze EC2 con i seguenti requisiti:	✓	✓
<ul style="list-style-type: none"> Le regole in uscita che consentono il traffico verso 0.0.0.0/0 (impostazione predefinita). 	✓	✓

Prerequisito	Per la piattaforma ECS	Per docker-local
<ul style="list-style-type: none">• Una regola in entrata che consente tutto il traffico da se stessa (lo stesso gruppo di sicurezza).	✓	✓
<ul style="list-style-type: none">• Una regola in entrata che consente tutto il traffico proveniente dal gruppo di sicurezza per l'Application Load Balancer.	✓	
<ul style="list-style-type: none">• Regole in entrata che consentono HTTP (porta 80), EthStats (servito sulla porta 8080), JSON RPC su HTTP (porta 8545) e SSH (porta 22) da fonti esterne affidabili, come l'IP CIDR del computer client.		✓

Prerequisito	Per la piattaforma ECS	Per docker-local
<p>Un gruppo di sicurezza per Application Load Balancer con i seguenti requisiti:</p> <ul style="list-style-type: none">• Una regola in entrata che consente tutto il traffico da se stessa (lo stesso gruppo di sicurezza).• Una regola in entrata che consente l'instradamento di tutto il traffico dal gruppo di sicurezza per le istanze EC2.• Le regole in uscita che consentono l'instradamento di tutto il traffico solo verso il gruppo di sicurezza per le istanze EC2. Per ulteriori informazioni, consulta Creazione di gruppi di sicurezza.• Se si associa lo stesso gruppo di sicurezza a un bastion host, una regola in entrata che consente il traffico SSH (porta 22) da origini attendibili.• Se il bastion host o un altro componente connesso a Internet si trova in un gruppo di sicurezza diverso, una regola in entrata che consente il traffico da tale componente.	✓	

Prerequisiti per VPC

Prerequisito	Per la piattaforma ECS	Per docker-local
Un indirizzo IP elastico, utilizzato per accedere ai servizi Ethereum.	✓	✓
Una sottorete per eseguire le istanze EC2. Ti consigliamo una sottorete privata.	✓	✓
Due sottoreti pubblicamente accessibili. Ogni sottorete deve trovarsi in zone di disponibilità diverse l'una dall'altra, una delle quali nella stessa zona di disponibilità della sottorete per le istanze EC2.	✓	

Esempi di autorizzazioni IAM per il profilo di istanza EC2 e il ruolo ECS

Quando utilizzi il modello, puoi specificare un ARN del profilo dell'istanza EC2 come uno dei parametri. Se utilizzi la piattaforma dei container ECS, puoi anche specificare un ARN del ruolo ECS. Le policy delle autorizzazioni collegate a questi ruoli consentono alle risorse AWS e alle istanze presenti nel cluster di interagire con altre risorse AWS. Per ulteriori informazioni, consulta [Ruoli IAM](#) nella Guida per l'utente di IAM. Utilizza le istruzioni e le procedure delle policy riportate di seguito come punto di partenza per la creazione delle autorizzazioni.

Esempio di policy delle autorizzazioni per il profilo dell'istanza EC2

Le seguenti policy delle autorizzazioni mostrano le operazioni consentite per il profilo dell'istanza EC2 quando scegli la piattaforma dei container ECS. Le stesse istruzioni delle policy possono essere utilizzate in una piattaforma di container docker-local, in cui le chiavi di contesto `ecs` sono state rimosse per limitare l'accesso.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ecs:CreateCluster",
        "ecs:DeregisterContainerInstance",
        "ecs:DiscoverPollEndpoint",
        "ecs:Poll",
        "ecs:RegisterContainerInstance",
        "ecs:StartTelemetrySession",
        "ecs:Submit*",
        "ecr:GetAuthorizationToken",
        "ecr:BatchCheckLayerAvailability",
        "ecr:GetDownloadUrlForLayer",
        "ecr:BatchGetImage",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "dynamodb:BatchGetItem",
        "dynamodb:BatchWriteItem",
        "dynamodb:PutItem",
        "dynamodb>DeleteItem",
        "dynamodb:GetItem",
        "dynamodb:Scan",
        "dynamodb:Query",
        "dynamodb:UpdateItem"
      ],
      "Resource": "*"
    }
  ]
}

```

Creazione del ruolo ECS e delle autorizzazioni

Per le autorizzazioni associate al ruolo ECS, ti consigliamo di iniziare con la politica di autorizzazione di Amazon EC2. ContainerServiceRole Utilizza la procedura seguente per creare un ruolo e collegare questa policy delle autorizzazioni. Utilizza la console IAM per visualizzare la maggior parte delle autorizzazioni in questa politica. up-to-date

Per creare il ruolo IAM per Amazon ECS

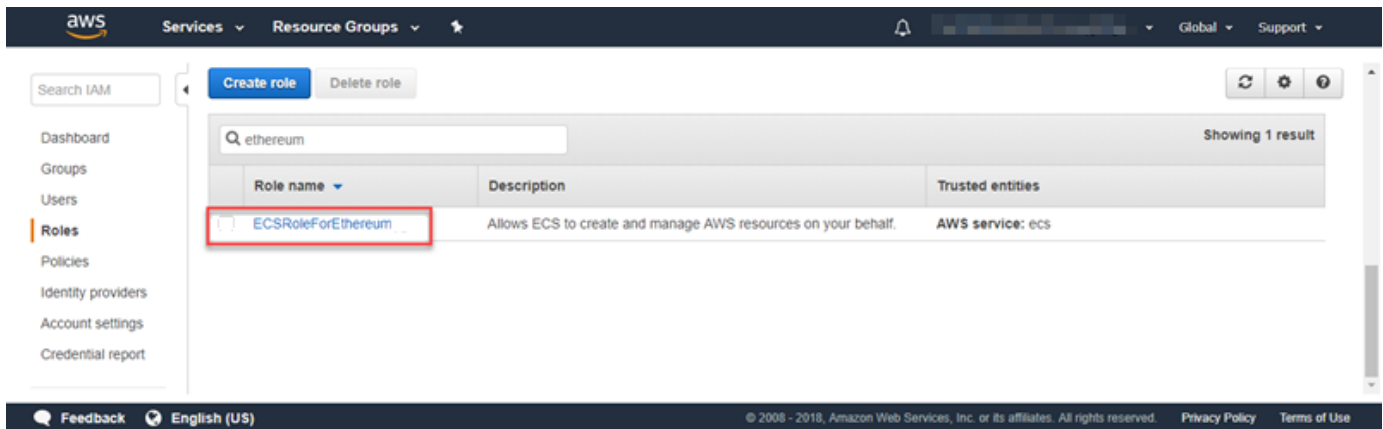
1. Aprire la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.

2. Nel riquadro di navigazione, seleziona Ruoli > Crea ruolo.
3. In Select type of trusted entity (Seleziona tipo di entità attendibile), scegli AWS service (Servizio AWS).
4. In Choose the service that will use this role (Scegli il servizio che utilizzerà questo ruolo), selezionare Elastic Container Service.
5. In Select your use case (Seleziona il tuo caso d'uso), seleziona Elastic Container Service, Next:Permissions.

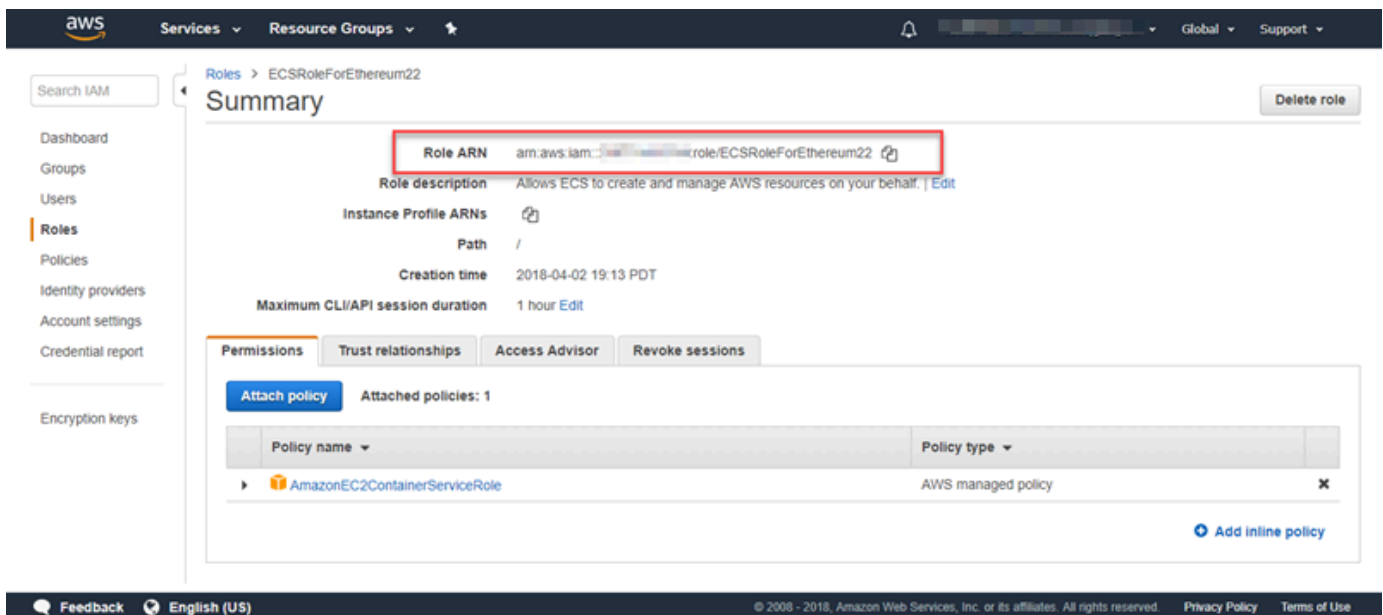
The screenshot shows the AWS IAM console 'Create role' wizard. The 'Select type of trusted entity' step has 'AWS service' selected. The 'Choose the service that will use this role' step shows a grid of services with 'Elastic Container Service' highlighted in a red box. The 'Select your use case' step shows 'Elastic Container Service' selected in a red box. The 'Next: Permissions' button is visible at the bottom right.

6. Per la politica di autorizzazione, lascia selezionata la politica predefinita (Amazon EC2 ContainerServiceRole) e scegli Avanti:Revisione.
7. Per Nome del ruolo, inserisci un valore che ti aiuti a identificare il ruolo, ad esempio ECS. RoleForEthereum Per Role Description (Descrizione ruolo) immetti un breve riepilogo. Annota il nome del ruolo per utilizzarlo in seguito.
8. Scegli Crea ruolo.

9. Dall'elenco, scegli il ruolo che hai appena creato. Se il tuo account ha molti ruoli, puoi cercare il nome del ruolo.



10. Copia il valore di Role ARN (ARN ruolo) e salvalo in modo che sia possibile copiarlo di nuovo. Hai bisogno di questo ARN quando crei la rete Ethereum.



Connessione a Ethereum Resources

Dopo che lo stack principale creato con il modello mostra CREATE_COMPLETE, puoi connetterti alle risorse Ethereum utilizzando la console. AWS CloudFormation La modalità di connessione dipende dalla piattaforma dei container (ECS o docker-local) che hai scelto:

- ECS: la scheda Output dello stack principale fornisce collegamenti ai servizi in esecuzione su Application Load Balancer. Per motivi di sicurezza, questi URL non sono accessibili direttamente.

Per connetterti, puoi configurare e utilizzare un bastion host per le connessioni proxy. Per ulteriori informazioni, consulta [Connessioni proxy utilizzando un host Bastion](#) di seguito.

- `docker-local`: ti connetti utilizzando l'indirizzo IP dell'istanza EC2 che ospita i servizi Ethereum come elencato di seguito. Utilizza la console EC2 per trovare l'indirizzo *ec2-IP-address* dell'istanza creata dal modello.
 - EthStats—Usa l'indirizzo IP `http://EC2`
 - EthExplorer—Usa l'indirizzo IP `http://EC2:8080`
 - EthJsonRpc—Usa l'indirizzo IP `http://EC2:8545`

Se hai specificato una sottorete pubblica per Ethereum Network Subnet ID (ID sottorete rete Ethereum) (List of VPC Subnets to use (Elenco sottoreti VPC da utilizzare) all'interno del modello), puoi connetterti direttamente. Il client deve essere un'origine di traffico in entrata affidabile per SSH (porta 22) e per le porte elencate. Questo è determinato dal gruppo di sicurezza EC2 che hai specificato utilizzando il modello AWS Blockchain per Ethereum.

Se hai specificato una sottorete privata, puoi impostare e utilizzare un bastion host per le connessioni proxy a tali indirizzi. Per ulteriori informazioni, consulta [Connessioni proxy utilizzando un host Bastion](#) di seguito.

Connessioni proxy utilizzando un host Bastion

Con alcune configurazioni, i servizi Ethereum potrebbero non essere disponibili al pubblico. In questi casi, puoi connetterti alle risorse di Ethereum tramite un host bastion. Per ulteriori informazioni sui bastion host, consulta [Architettura dei bastion host di Linux](#) nella Guida rapida sui bastion host di Linux.

L'host bastion è un'istanza EC2. Assicurati che siano soddisfatti i seguenti requisiti:

- L'istanza EC2 per l'host bastion si trova all'interno di una sottorete pubblica con Auto-Assign Public IP abilitato e che dispone di un gateway Internet.
- L'host bastion ha la key pair che consente le connessioni ssh.
- L'host bastion è associato a un gruppo di sicurezza che consente il traffico SSH in entrata dai client che si connettono.
- Il gruppo di sicurezza assegnato agli host Ethereum (ad esempio, l'Application Load Balancer se ECS è la piattaforma container, o l'istanza EC2 host se `docker-local` è la piattaforma container) consente il traffico in entrata su tutte le porte da sorgenti all'interno del VPC.

Con un bastion host configurato, assicurati che i client che si connettono utilizzino l'host bastion come proxy. L'esempio seguente mostra la configurazione di una connessione proxy utilizzando Mac OS. Sostituisci *BastionIP* con l'indirizzo IP dell'istanza EC2 del bastion host e *MySshKey.pem* con il file key pair che hai copiato sull'host bastion.

Nella riga di comando, digita quanto segue:

```
ssh -i mySshKey.pem ec2-user@BastionIP -D 9001
```

Questo imposta il port forwarding per la porta 9001 sul computer locale verso l'host bastion.

Successivamente, configura il browser o il sistema per utilizzare il proxy SOCKS per. localhost:9001 Ad esempio con Mac OS, seleziona System Preferences (Preferenze di sistema), Network (Rete), Advanced (Avanzate), seleziona proxy SOCKS e digita localhost:9001.

Utilizzando FoxyProxy Standard con Chrome, seleziona Altri strumenti, Estensioni. In FoxyProxy Standard, seleziona Dettagli, Opzioni di estensione, Aggiungi nuovo proxy. Seleziona Manual Proxy Configuration (Configurazione proxy manuale). In Host or IP Address (Indirizzo IP o host), digita localhost e in Port (Porta), digita 9001. Seleziona SOCKS proxy?, Salva.

Ora dovresti essere in grado di connetterti agli indirizzi host di Ethereum elencati nell'output del modello.

Utilizzo del modello AWS Blockchain per Hyperledger Fabric

Hyperledger Fabric è un framework blockchain che esegue contratti intelligenti chiamati chaincode, scritti in Go. Puoi creare una rete privata con Hyperledger Fabric, limitando il numero di peer che possono connettersi e partecipare alla rete. [Per ulteriori informazioni su Hyperledger Fabric, consulta la documentazione di Hyperledger Fabric.](#) [Per ulteriori informazioni su chaincode, consulta l'argomento Chaincode for Developers nella documentazione di Hyperledger Fabric.](#)

Il modello AWS Blockchain per Hyperledger Fabric supporta solo una piattaforma container docker locale, il che significa che i contenitori Hyperledger Fabric vengono distribuiti su una singola istanza EC2.

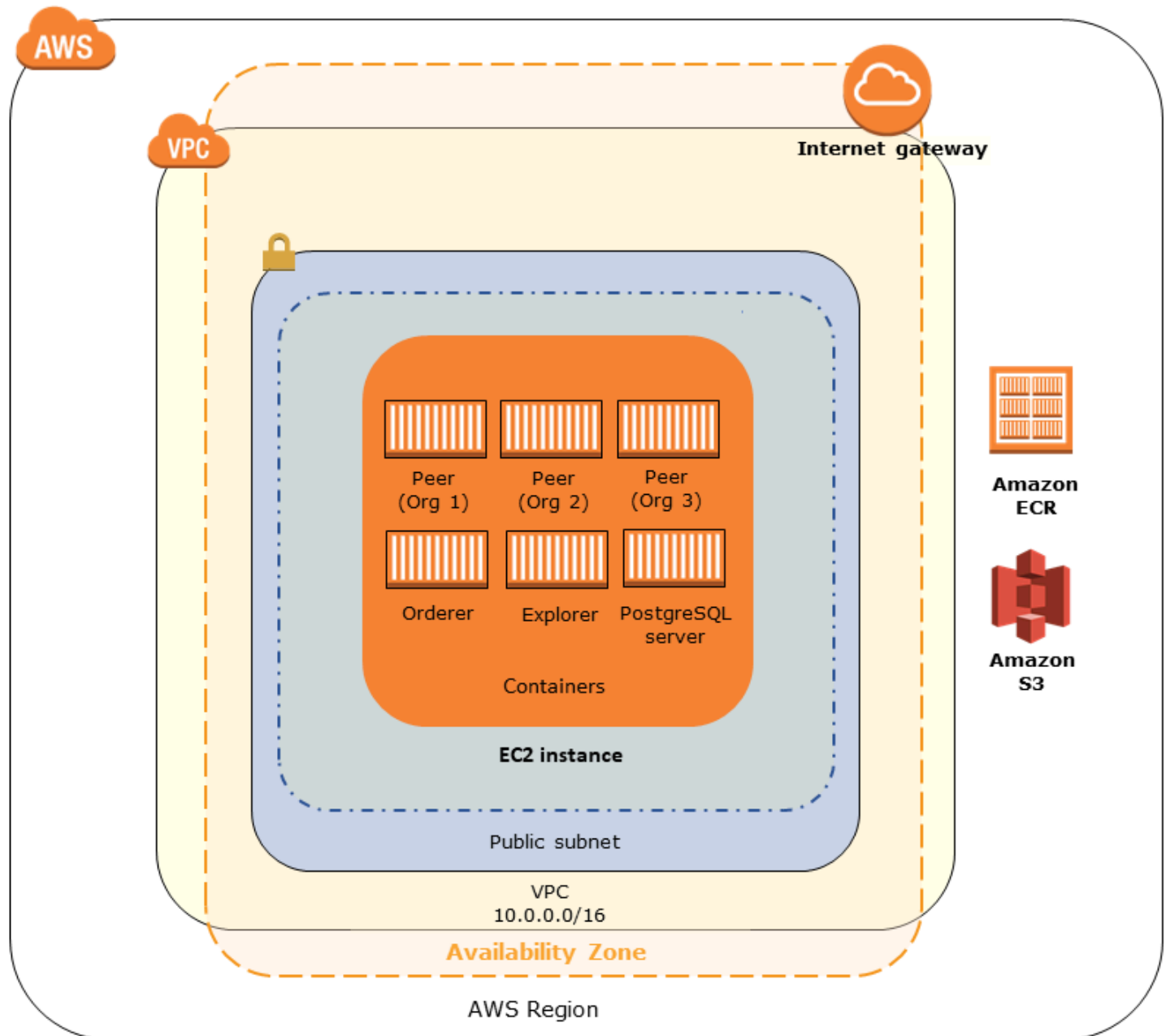
Link per l'avvio

Consulta [Getting Started with AWS Blockchain Templates](#) per i link da avviare AWS CloudFormation in regioni specifiche utilizzando i modelli Hyperledger Fabric.

Modello AWS Blockchain per componenti Hyperledger Fabric

Il modello AWS Blockchain per Hyperledger Fabric crea un'istanza EC2 con Docker e avvia una rete Hyperledger Fabric utilizzando contenitori su quell'istanza. La rete include un servizio di gestione degli ordini e tre organizzazioni, ognuna con un servizio peer. Il modello inoltre avvia un container Hyperledger Explorer, che consente di esplorare i dati della blockchain. Per supportare Hyperledger Explorer, viene avviato un container del server PostgreSQL.

Il diagramma seguente mostra una rete Hyperledger Fabric creata utilizzando il modello:



Prerequisiti

Prima di avviare una rete Hyperledger Fabric utilizzando un modello, assicurati che siano soddisfatti i seguenti requisiti:

- Il principio IAM (utente o gruppo) che utilizzi deve disporre dell'autorizzazione per funzionare con tutti i servizi correlati.
- È necessario avere accesso a una coppia di chiavi utilizzabile per accedere alle istanze EC2, ad esempio tramite SSH. La chiave deve esistere nella stessa regione dell'istanza.
- È necessario disporre di un profilo di istanza EC2 con una politica di autorizzazioni allegata che consenta l'accesso ad Amazon S3 e ad Amazon Elastic Container Registry (Amazon ECR) per estrarre i contenitori. Per un esempio di policy delle autorizzazioni, consulta [Esempi di autorizzazioni IAM per il profilo di istanza EC2](#).
- È necessario disporre di una rete Amazon VPC con una sottorete pubblica o una sottorete privata con un gateway NAT e un indirizzo IP elastico per poter accedere ad Amazon S3 AWS CloudFormation e Amazon ECR.
- È necessario disporre di un gruppo di sicurezza EC2 con regole in entrata che consentono il traffico SSH (porta 22) dagli indirizzi IP che devono connettersi all'istanza tramite SSH. Lo stesso requisito è valido per i client che devono connettersi a Hyperledger Explorer (porta 8080).

Esempi di autorizzazioni IAM per il profilo di istanza EC2

Quando usi AWS Blockchain Template per Hyperledger Fabric, specifichi un ARN del profilo di istanza EC2 come uno dei parametri. Utilizza la seguente istruzione come punto di partenza per la policy delle autorizzazioni collegata al profilo dell'istanza e del ruolo EC2 in questione.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ecr:GetAuthorizationToken",
        "ecr:BatchCheckLayerAvailability",
        "ecr:GetDownloadUrlForLayer",
        "ecr:GetRepositoryPolicy",
        "ecr:DescribeRepositories",
```

```
        "ecr:ListImages",
        "ecr:DescribeImages",
        "ecr:BatchGetImage",
        "s3:Get*",
        "s3:List*"
    ],
    "Resource": "*"
}
]
```

Connessione alle risorse di Hyperledger Fabric

Dopo che lo stack principale creato con il modello mostra `CREATE_COMPLETE`, puoi connetterti alle risorse Hyperledger Fabric sull'istanza EC2. Se hai specificato una sottorete pubblica, puoi connetterti a tale istanza seguendo la stessa procedura utilizzata per qualsiasi altra istanza EC2. Per ulteriori informazioni, consulta [Connessione all'istanza Linux tramite SSH nella Guida per l'utente](#) di Amazon EC2.

Se hai specificato una sottorete privata, puoi configurare e utilizzare un bastion host per eseguire il proxy delle connessioni alle risorse Hyperledger Fabric. Per ulteriori informazioni, consulta [Connessioni proxy utilizzando un host Bastion](#) di seguito.

Note

Potresti notare che il modello alloca un indirizzo IP pubblico all'istanza EC2 che ospita i servizi Hyperledger Fabric; tuttavia, questo indirizzo IP non è accessibile al pubblico perché le politiche di routing nella sottorete privata specificata non consentono il traffico tra questo indirizzo IP e le fonti pubbliche.

Connessioni proxy che utilizzano un host Bastion

Con alcune configurazioni, i servizi Hyperledger Fabric potrebbero non essere disponibili al pubblico. In questi casi, puoi connetterti alle risorse di Hyperledger Fabric tramite un host bastion. Per ulteriori informazioni sui bastion host, consulta [Architettura dei bastion host di Linux](#) nella Guida rapida sui bastion host di Linux.

L'host bastion è un'istanza EC2. Assicurati che siano soddisfatti i seguenti requisiti:

- L'istanza EC2 per l'host bastion si trova all'interno di una sottorete pubblica con Auto-Assign Public IP abilitato e che dispone di un gateway Internet.
- L'host bastion ha la key pair che consente le connessioni ssh.
- L'host bastion è associato a un gruppo di sicurezza che consente il traffico SSH in entrata dai client che si connettono.
- Il gruppo di sicurezza assegnato agli host Hyperledger Fabric (ad esempio, Application Load Balancer se ECS è la piattaforma container, o l'istanza EC2 host se docker-local è la piattaforma container) consente il traffico in entrata su tutte le porte da sorgenti all'interno del VPC.

Con un bastion host configurato, assicurati che i client che si connettono utilizzino l'host bastion come proxy. L'esempio seguente mostra la configurazione di una connessione proxy utilizzando Mac OS. Sostituisci *BastionIP* con l'indirizzo IP dell'istanza EC2 del bastion host e *MySshKey.pem* con il file key pair che hai copiato sull'host bastion.

Nella riga di comando, digita quanto segue:

```
ssh -i mySshKey.pem ec2-user@BastionIP -D 9001
```

Questo imposta il port forwarding per la porta 9001 sul computer locale verso l'host bastion.

Successivamente, configura il browser o il sistema per utilizzare il proxy SOCKS per. localhost:9001 Ad esempio con Mac OS, seleziona System Preferences (Preferenze di sistema), Network (Rete), Advanced (Avanzate), seleziona proxy SOCKS e digita localhost:9001.

Utilizzando FoxyProxy Standard con Chrome, seleziona Altri strumenti, Estensioni. In FoxyProxy Standard, seleziona Dettagli, Opzioni di estensione, Aggiungi nuovo proxy. Seleziona Manual Proxy Configuration (Configurazione proxy manuale). In Host or IP Address (Indirizzo IP o host), digita localhost e in Port (Porta), digita 9001. Seleziona SOCKS proxy?, Salva.

Ora dovresti essere in grado di connetterti agli indirizzi host di Hyperledger Fabric elencati nell'output del modello.

Cronologia dei documenti

La tabella seguente descrive le modifiche apportate alla documentazione per questa guida.

Ultimo aggiornamento della documentazione: 1 maggio 2019

Modifica	Descrizione	Data
Interruzione di AWS Blockchain Templates.	AWS Blockchain Templates è stato interrotto il 30 aprile 2019. Non verranno effettuati ulteriori aggiornamenti a questo servizio o alla presente documentazione di supporto. Per la migliore esperienza di Managed Blockchain su AWS, ti consigliamo di utilizzare Amazon Managed Blockchain (AMB) .	1 maggio 2019
Aggiornamenti del bastion host.	Modificato il tutorial sulle nozioni di base e requisiti relativi ai prerequisiti di Ethereum per l'aggiunta di un bastion host, che consente l'accesso alle risorse Web servite attraverso il sistema di bilanciamento del carico interno quando si utilizza la piattaforma ECS e l'istanza EC2 quando si utilizza docker-local.	3 maggio 2018
È stata creata la guida.	Nuova guida per sviluppatori per supportare la versione iniziale di AWS Blockchain Templates.	19 aprile 2018

Glossario AWS

Per la terminologia AWS più recente, consultare il [glossario AWS](#) nella documentazione di riferimento per Glossario AWS.