



开发人员指南

# AWS Blockchain Templates



# AWS Blockchain Templates: 开发人员指南

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon 的商标和商业外观不得用于任何非 Amazon 的商品或服务，也不得以任何可能引起客户混淆、贬低或诋毁 Amazon 的方式使用。所有非 Amazon 拥有的其他商标均为各自所有者的财产，这些所有者可能附属于 Amazon、与 Amazon 有关联或由 Amazon 赞助，也可能不是如此。

# Table of Contents

.....	iv
什么是 AWS Blockchain Templates ? .....	1
如何开始 .....	2
我精通区块 AWS 链 .....	2
我精通区块链，对区块 AWS 链不熟悉 .....	3
我是区块链的初学者 AWS，精通区块链 .....	3
我是区块链的新手 AWS .....	3
相关服务 .....	3
设置 .....	4
注册 AWS .....	4
创建 IAM 用户 .....	5
创建密钥对 .....	6
开始使用 .....	8
设置先决条件 .....	9
创建 VPC 和子网 .....	9
创建安全组 .....	12
创建适用于 Amazon ECS 的 IAM 角色和 EC2 实例配置文件 .....	14
创建堡垒主机 .....	19
创建 Ethereum 网络 .....	20
连接 EthStats 并 EthExplorer 使用堡垒主机 .....	23
清理资源 .....	26
Amazon Web Services Blockchain Templates 和特征 .....	27
适用于 Ethereum 的 AWS Blockchain Templates .....	27
启动链接 .....	27
Ethereum 选项 .....	27
先决条件 .....	31
连接到 Ethereum 资源 .....	37
适用于 Hyperledger Fabric 的 AWS Blockchain Templates .....	39
启动链接 .....	39
适用于 Hyperledger Fabric 组件的 Amazon Web Services Blockchain Templates .....	39
先决条件 .....	40
连接到 Hyperledger Fabric Resources .....	42
文档历史记录 .....	44
AWS 词汇表 .....	45

Amazon Web Services Blockchain Templates 已于 2019 年 4 月 30 日停产。不会对本服务或本支持文档进行进一步更新。为了获得最佳的托管区块链体验 AWS，我们建议您使用[亚马逊托管区块链 \(AMB\)](#)。要了解有关 Amazon Managed Blockchain 入门的更多信息，请参阅[Hyperledger Fabric 研讨会或关于部署 Ethereum 节点的博客](#)。如果您对 AMB 有疑问或需要进一步支持，[请联系支持](#)或您的 AWS 客户团队。

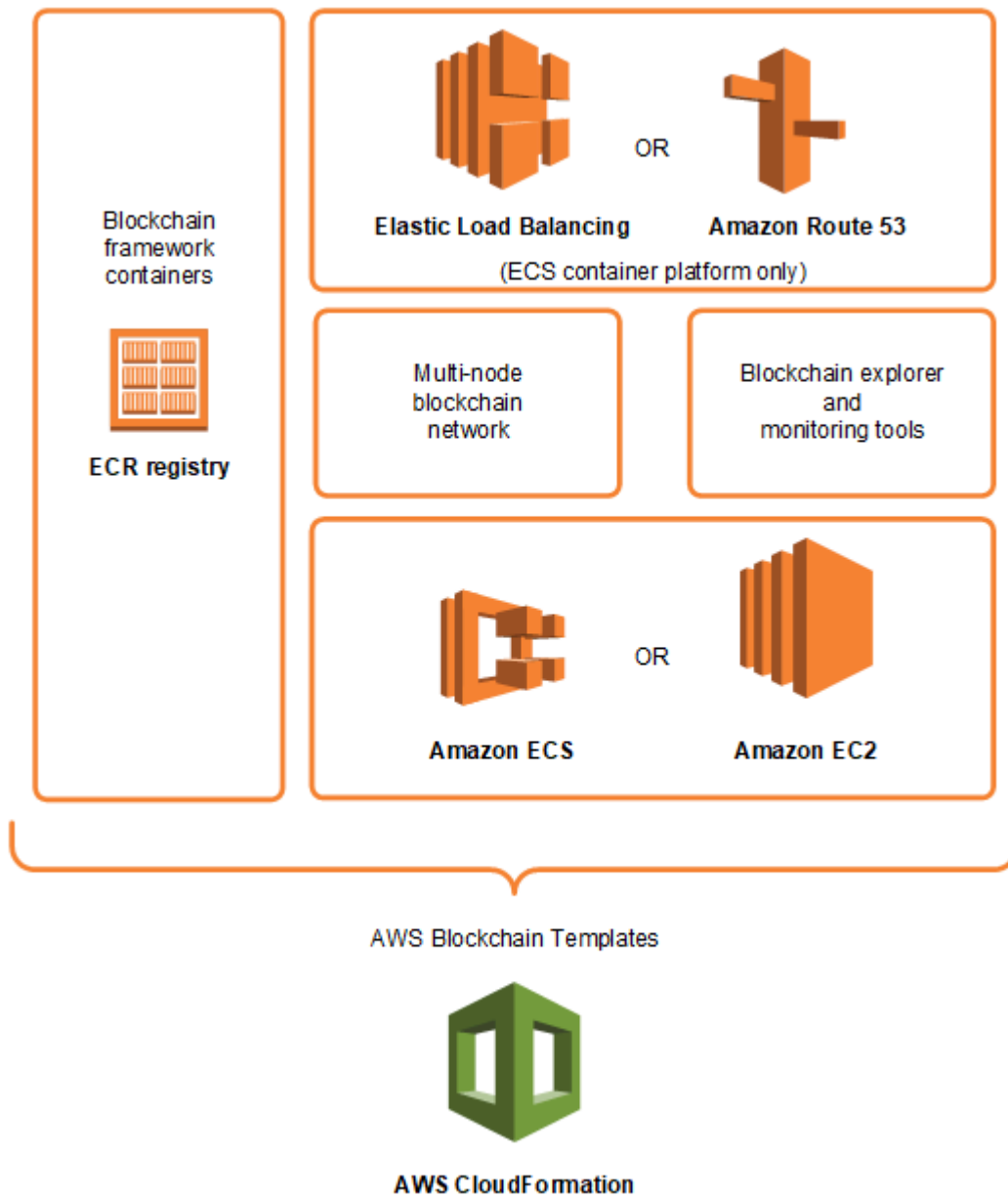
本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。

# 什么是 AWS Blockchain Templates ?

AWS Blockchain Templates 可帮助您 AWS 使用不同的区块链框架快速创建和部署区块链网络。区块链是一种分散式数据库技术，用于维护不断增长的事务和智能合同集合，这些集合使用加密防止篡改和修订。

区块链网络是一种 peer-to-peer 网络，可提高国际支付、供应链管理、土地登记、众筹、治理、金融交易等业务流程的交易效率和不可改变性。这能让彼此不认识的人员和组织互相信任和独立验证事务记录。

您可以使用 AWS Blockchain Templates 来配置和启动 CloudFormation 堆栈以创建区块链网络。您使用的 AWS 资源和服务取决于您选择的 AWS Blockchain Template 和指定的选项。有关可用模板及其功能的信息，请参阅 [Amazon Web Services Blockchain Templates 和特征](#)。下图显示了使用 AWS Blockchain Templates AWS 创建的区块链网络的基本组件。



## 如何开始

最佳起点取决于您在区块链方面的专业水平，AWS尤其是与 AWS Blockchain Templates 相关的服务。

## 我精通区块 AWS 链

从 [Amazon Web Services Blockchain Templates 和特征](#) 中的主题开始，了解您希望使用的框架。使用链接启动 AWS Blockchain Template 并配置区块链网络，或下载模板并亲自查看。

## 我精通区块链，对区块 AWS 链不熟悉

从 [Amazon Web Services Blockchain Templates 入门](#) 教程开始学习。该教程引导您利用默认设置创建入门级 Ethereum 区块链网络。完成后，请参阅 [Amazon Web Services Blockchain Templates 和特征](#)，了解区块链框架概述，并通过链接进一步了解配置选项和功能。

## 我是区块链的初学者 AWS，精通区块链

首先是[设置 Amazon Web Services Blockchain Templates](#)。这可以帮助您设置基本知识 AWS，例如帐户和用户个人资料。接下来学习 [Amazon Web Services Blockchain Templates 入门](#) 教程。本教程引导您创建入门级 Ethereum 区块链网络。即使您最终不使用 Ethereum，也可获得设置相关服务的实践经验。这种体验对于所有区块链框架都很有用。最后，请参阅 [Amazon Web Services Blockchain Templates 和特征](#) 部分有关您所用框架的主题。

## 我是区块链的新手 AWS

首先是[设置 Amazon Web Services Blockchain Templates](#)。这可以帮助您设置基本知识 AWS，例如帐户和用户个人资料。接下来学习 [Amazon Web Services Blockchain Templates 入门](#) 教程。本教程引导您创建入门级 Ethereum 区块链网络。花点时间浏览链接，详细了解 AWS 服务和以太坊。

## 相关服务

根据您的选择的选项，AWS Blockchain Templates 可以使用以下 AWS 服务来部署区块链：

- Amazon EC2—为您的区块链网络提供计算容量。有关更多信息，请参阅 [《Amazon EC2 用户指南》](#)。
- Amazon ECS—如果您选择使用区块链网络，则在集群中的多个 EC2 实例之间为您的网络计划容器部署。有关更多信息，请参阅 [Amazon Elastic Container Service 开发人员指南](#)。
- Amazon VPC—为您创建的 Ethereum 资源提供网络访问权限。您可以自定义可访问性和安全性的配置。有关更多信息，请参阅 [Amazon VPC 开发人员指南](#)。
- 应用程序负载均衡—在将 Amazon ECS 作为容器平台时，作为单一接触点以访问可用的用户接口和内部服务发现。有关更多信息，请参阅应用程序负载均衡器用户指南中的[应用程序负载均衡器的配额](#)。

# 设置 Amazon Web Services Blockchain Templates

开始使用 AWS Blockchain Templates 之前，请完成以下任务：

- [注册 AWS](#)
- [创建 IAM 用户](#)
- [创建密钥对](#)

这些是所有区块链配置的基本先决条件。此外，您选择的区块链网络可能另有先决条件，取决于所需的环境和配置选项。有关更多信息，请参阅 [Amazon Web Services Blockchain Templates 和特征](#) 中您所选区块链模板的相关部分。

有关使用 Amazon ECS 集群为以太坊私有网络设置先决条件的 step-by-step 说明，请参阅 [Amazon Web Services Blockchain Templates 入门](#)。

## 注册 AWS

当您注册时 AWS，您的 AWS 帐户会自动注册所有服务。您只需为使用的服务付费。

如果您已经有一个 AWS 帐户，请跳到下一个任务。如果您还没有 AWS 帐户，请按照以下步骤创建。

### 创建 AWS 账户

1. 打开 <https://portal.aws.amazon.com/billing/注册>。
2. 按照屏幕上的说明操作。

在注册时，将接到电话或收到短信，要求使用电话键盘输入一个验证码。

当您注册时 AWS 账户，就会创建 AWS 账户根用户一个。根用户有权访问该账户中的所有 AWS 服务和资源。作为最佳安全实践，请为用户分配管理访问权限，并且只使用根用户来执行 [需要根用户访问权限的任务](#)。

记下您的 AWS 账号。在下一个任务中创建 IAM 用户时，您将需要它。

## 创建 IAM 用户

中的服务 AWS 要求您在访问时提供证书，以便服务可以确定您是否有权访问其资源。控制台要求您的密码。您可以为 AWS 账户创建访问密钥以访问命令行界面或 API。但是，我们不建议您 AWS 使用 AWS 账户的证书进行访问；我们建议您改用 AWS Identity and Access Management (IAM)。创建 IAM 用户，然后将该用户添加到具有管理权限的 IAM 组或授予此用户管理权限。然后，您可以使用特殊的 URL 和 IAM 用户的证书进行访问 AWS。

如果您已注册 AWS 但尚未为自己创建 IAM 用户，则可以使用 IAM 控制台创建一个。如果您已有 IAM 用户，则可跳过本步骤。

要创建管理员用户，请选择以下选项之一。

选择一种方法来管理您的管理员	目标	方式	您也可以
在 IAM Identity Center 中 (推荐)	使用短期凭证访问 AWS。  这符合安全最佳实操。有关最佳实践的信息，请参阅《IAM 用户指南》中的 <a href="#">IAM 中的安全最佳实践</a> 。	有关说明，请参阅《AWS IAM Identity Center 用户指南》中的 <a href="#">入门</a> 。	通过在《AWS Command Line Interface 用户指南》 <a href="#">AWS IAM Identity Center 中配置 AWS CLI 要使用的来配置编程访问权限</a> 。
在 IAM 中 (不推荐使用)	使用长期凭证访问 AWS。	按照《IAM 用户指南》中的 <a href="#">创建用于紧急访问的 IAM 用户</a> 中的说明进行操作。	按照《IAM 用户指南》中的 <a href="#">管理 IAM 用户的访问密钥</a> ，配置程式访问。

要以此新 IAM 用户的身份登录，请注销，然后使用以下 URL AWS 管理控制台，其中 `your_aws_account_id` 是您的 AWS 账号，不带连字符（例如，如果您的账号为 `1234-5678-9012`，则您的 AWS 账户 ID 为 `123456789012`）：

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

输入您刚创建的 IAM 用户名和密码。登录后，导航栏显示“*your\_user\_name @ your\_aws\_account\_id*”。

如果您不希望登录页面的 URL 包含您的 AWS 账户 ID，则可以创建账户别名。从 IAM 控制面板中，选择“创建账户别名”，然后输入一个别名，例如您的公司名称。要在创建账户别名后登录，请使用以下 URL：

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

要为您的账户验证 IAM 用户的登录链接，请打开 IAM 控制台并在控制面板的 IAM 用户登录链接下进行检查。

有关更多信息，请参阅 [AWS Identity and Access Management 用户指南](#)。

## 创建密钥对

AWS 使用公钥加密来保护区块链网络中实例的登录信息。指定您在使用每个 AWS Blockchain Template 时的密钥对名称。然后，您可以使用该密钥对直接访问实例，例如使用 SSH 登录。

如果您已有正确区域中的密钥对，可跳过本步骤。如果您尚未创建密钥对，则可以使用 Amazon EC2 控制台创建密钥对。在用于启动 Ethereum 网络的区域中创建密钥对。有关更多信息，请参阅 Amazon EC2 用户指南中的 [区域和可用区](#)。

### 创建密钥对

1. 打开 Amazon EC2 控制台，网址为 <https://console.aws.amazon.com/ec2/>。
2. 从导航栏中，选择密钥对所在的区域。您可以选择可用的任意区域，无论您的位置如何；但是，密钥对是该区域特定的。例如，如果您计划在美国东部（俄亥俄州）区域中启动实例，则必须在相同区域中为实例创建密钥对。
3. 在导航窗格中，选择 Key Pairs 和 Create Key Pair。
4. 对于 Key pair name (密钥对名称)，输入新密钥对的名称。选择一个容易记住的名称，例如，您的 IAM 用户名，后跟 `-key-pair` 加区域名称。例如，`me-key-pair-useast2`。选择创建。
5. 您的浏览器会自动下载私有密钥文件。基本文件名是您为密钥对指定的名称，文件扩展名为 `.pem`。将私有密钥文件保存在安全位置。

**⚠ Important**

这是您保存私有密钥文件的唯一机会。在启动 Ethereum 网络时，请提供密钥对的名称。

有关更多信息，请参阅《[亚马逊 EC2 用户指南](#)》中的[亚马逊 EC2 密钥对](#)。有关使用密钥对连接 EC2 实例的更多信息，请参阅《[亚马逊 EC2 用户指南](#)》中的“[连接到您的 Linux 实例](#)”。

# Amazon Web Services Blockchain Templates 入门

本教程演示了如何使用适用于以太坊的 AWS 区块链模板来创建私有区块链网络 CloudFormation。AWS 您创建的网络有两个 Ethereum 客户端，和一个在 Amazon ECS 群集中的 Amazon EC2 实例上运行的挖掘程序。Amazon ECS 在从 Amazon ECR 拉出的 Docker 容器中运行此服务。在开始本教程之前，了解一下区块链网络和所涉及的 AWS 服务会很有帮助，但这不是必需的。

本教程假定您已设置好 [设置 Amazon Web Services Blockchain Templates](#) 中提到的一般先决条件。此外，在使用模板之前，您必须设置一些 AWS 资源，例如 Amazon VPC 网络和 IAM 角色的特定权限。

本教程演示如何设置这些先决条件。我们选择了一些设置，但它们不是规定性的。只要满足先决条件，您就可以根据应用程序和环境需求选择其他配置。有关每个模板的功能和一般先决条件以及下载模板或在 CloudFormation 中直接启动模板的信息，请参阅 [Amazon Web Services Blockchain Templates 和特征](#)。

本教程中的示例使用美国西部（俄勒冈州）区域（us-west-2），但您可以使用支持 AWS Blockchain Templates 的任何区域：

- 美国西部（俄勒冈州）区域（us-west-2）
- 美国东部（弗吉尼亚北部）区域（us-east-1）
- 美国东部（俄亥俄）区域（us-east-2）

## Note

在上述未列区域运行模板会在美国东部（弗吉尼亚州北部）区域（us-east-1）中启动资源。

您使用本教程配置的 AWS Blockchain Template 创建以下资源：

- 您指定的类型和数量的按需 EC2 实例。本教程使用默认的 t2.medium 实例类型。
- 内部应用程序负载均衡器。

在本程序后面提供了清理所创建资源的步骤。

主题

- [设置先决条件](#)

- [创建 Ethereum 网络](#)
- [连接 EthStats 并 EthExplorer 使用堡垒主机](#)
- [清理资源](#)

## 设置先决条件

您在本教程中指定的适用于 Ethereum 的 AWS Blockchain Template 配置要求您执行以下操作：

- [创建 VPC 和子网](#)
- [创建安全组](#)
- [创建适用于 Amazon ECS 的 IAM 角色和 EC2 实例配置文件](#)
- [创建堡垒主机](#)

## 创建 VPC 和子网

适用于 Ethereum 的 Amazon Web Services Blockchain Templates 将资源启动到使用 Amazon Virtual Private Cloud (Amazon VPC) 定义的虚拟网络中。您在本教程中指定的配置创建应用程序负载均衡器，它需要使用两个位于不同可用区中的公有子网。此外，容器实例需要使用一个私有子网，该子网必须与应用程序负载均衡器位于同一可用区中。首先，您使用 VPC 向导在同一可用区中创建一个公有子网和一个私有子网。然后，您在该 VPC 上的另一个可用区中创建第二个公有子网。

有关更多信息，请参阅《Amazon VPC 用户指南》中的[什么是 Amazon VPC ?](#)。

使用 Amazon VPC 控制台 (<https://console.aws.amazon.com/vpc/>) 创建弹性 IP 地址、VPC 和子网，如下所述。

### 创建弹性 IP 地址

1. 打开位于 <https://console.aws.amazon.com/vpc/> 的 Amazon VPC 控制台。
2. 选择“弹性”IPs、“分配新地址”、“分配”。
3. 记下您创建的弹性 IP 地址，然后选择 Close (关闭)。
4. 在弹性 IP 地址列表中，找到以前创建的弹性 IP 地址的 Allocation ID (分配 ID)。在创建 VPC 时会用到它。

## 创建 VPC

1. 在导航栏中，为 VPC 选择一个区域。VPCs 特定于某个区域，因此请选择您在其中创建密钥对以及启动以太坊堆栈的相同区域。有关更多信息，请参阅 [创建密钥对](#)。
2. 在 VPC 控制面板上，选择 Start VPC Wizard。
3. 在 Step 1: Select a VPC Configuration (步骤1: 选择 VPC 配置) 页面上，选择 VPC with Public and Private Subnets (具有公有子网和私有子网的 VPC)，然后选择 Select (选择)。
4. 在“步骤 2：包含公有和私有子网的 VPC”页面上，将 IPv4 CIDR 块和 IPv6 CIDR 块保留为默认值。对于 VPC name (VPC 名称)，输入一个易于理解的名称。
5. 对于公有子网的 IPv4 CIDR，请保留默认值。对于 Availability Zone (可用区)，选择一个区域。对于 Public subnet name (公有子网名称)，输入一个易于理解的名称。

如果使用模板，您可以将此子网指定为应用程序负载均衡器的两个子网中的第一个。

记下该子网的可用区，因为您为私有子网选择相同的可用区，并为另一个公有子网选择不同的可用区。

6. 对于私有子网的 IPv4 CIDR，请保留默认值。对于 Availability Zone (可用区)，选择与上一步相同的可用区。对于 Private subnet name (私有子网名称)，输入一个易于理解的名称。
7. 对于 Elastic IP Allocation ID (弹性 IP 分配 ID)，选择您以前创建的弹性 IP 地址。
8. 为其他设置保留默认值。
9. 选择创建 VPC。

以下示例显示了具有公有子网 EthereumPubSub1 和私有子网 1 的 EthereumNetworkV PC VPC。EthereumPvtSub公有子网使用 us-west-2a 可用区。

## 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:**\*  ▼

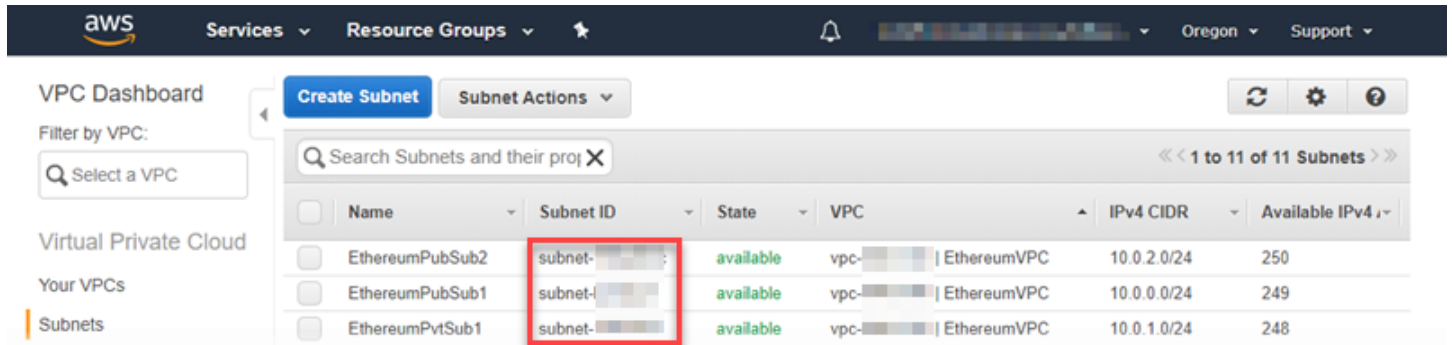
---

在其他可用区中创建第二个公有子网。

1. 选择 Subnets (子网)，然后从列表中选择以前创建的公有子网。选择 Route Table (路由表) 选项卡，并记下 Route table (路由表) ID。您为下面的第二个公有子网指定相同的路由表。
2. 选择创建子网。
3. 对于名称标签，输入子网的名称。稍后在该网络中创建堡垒主机时，您将使用该名称。

- 对于 VPC，选择您以前创建的 VPC。
- 对于 Availability Zone (可用区)，选择与第一个公有子网不同的可用区。
- 对于 IPv4 CIDR 区块，请输入 10.0.2.0/24。
- 选择是，创建。该子网将添加到子网列表中。
- 从列表中选择子网后，选择 Subnet Actions (子网操作)，然后选择 Modify auto-assign IP settings (修改自动分配 IP 设置)。选择“自动分配”、“保存” IPs、“关闭”。这样，在该子网中创建堡垒主机时，该主机可以获得一个公有 IP 地址。
- 在 Route Table (路由表) 选项卡上，选择 Edit (编辑)。对于 Change to (更改为)，选择您以前记下的路由表 ID，然后选择 Save (保存)。

您现在应该可以看到之前创建的 VPC 的三个子网。记下子网名称，IDs 以便您可以使用模板指定它们。



## 创建安全组

安全组可充当防火墙，控制资源的入站和出站流量。当您使用模板在 Amazon ECS 集群上创建 Ethereum 网络时，需指定两个安全组：

- 一个安全组适用于 EC2 实例，用于控制进出集群中的 EC2 实例的流量
- 应用程序负载均衡器的安全组，用于控制应用程序负载均衡器、EC2 实例和堡垒主机之间的流量。您也将该安全组与堡垒主机相关联。

每个安全组中都有允许应用程序负载均衡器和 EC2 实例之间进行通信的规则，以及其他最低要求规则。这就要求安全组相互引用。因此，您可以首先创建安全组，然后用适当的规则进行更新。

### 创建两个安全组

- 打开位于 <https://console.aws.amazon.com/ec2/> 的 Amazon EC2 控制台。

2. 在导航窗格中，选择 Security Groups (安全组)，然后选择 Create Security Group (创建安全组)。
3. 在安全组名称中，输入一个易于识别且能与其他安全组区分开来的名称，例如以太坊 EC2-SG 或 Ethereumalb-SG。稍后会用到这些名称。对于 Description (描述)，输入简短摘要。
4. 对于 VPC，选择您以前创建的 VPC。
5. 选择创建。
6. 重复以上步骤创建其他安全组。

### 在 EC2 实例的安全组中添加入站规则

1. 选择您之前创建的 EC2 实例的安全组
2. 在 Inbound (入站) 选项卡上，选择 Edit (编辑)。
3. 对于类型，请选择所有流量。对于“来源”，选择“自定义”，然后从列表中选择您当前正在编辑的安全组，例如 Ethereum EC2-SG。这样安全组中的 EC2 实例就可以相互通信。
4. 选择添加规则。
5. 对于 Type (类型)，请选择 All traffic (所有流量)。对于 Source (来源)，将 Custom (自定义) 保持选中状态，然后从列表中选择应用程序负载均衡器的安全组，例如 EthereumALB-SG。这样安全组中的 EC2 实例就可以与应用程序负载均衡器通信。
6. 选择保存。

### 针对应用程序负载均衡器的安全组添加入站规则并编辑出站规则

1. 选择您之前创建的应用程序负载均衡器安全组
2. 在 Inbound (入站) 选项卡上选择 Edit (编辑)，然后添加以下入站规则：
  - a. 对于类型，请选择所有流量。对于 Source (来源)，将 Custom (自定义) 保持选中状态，然后从列表中选择您当前编辑的安全组，例如 EthereumALB-SG。这允许应用程序负载均衡器与自身和堡垒主机进行通信。
  - b. 选择添加规则。
  - c. 对于 Type (类型)，请选择 All traffic (所有流量)。对于“源”，选择“自定义”，然后从列表中选择 EC2 实例的安全组，例如 Ethereum EC2-SG。这允许安全组中的 EC2 实例与应用程序负载均衡器和堡垒主机进行通信。
  - d. 选择添加规则。

- e. 对于 Type，选择 SSH。对于 Source (来源)，选择 My IP (我的 IP)，这会检测并输入您的计算机的 IP CIDR。

**⚠ Important**

该规则允许堡垒主机从您的计算机接受 SSH 流量，从而允许您的计算机使用堡垒主机查看 Web 界面并连接到 Ethereum 网络上的 EC2 实例。要允许其他计算机连接到 Ethereum 网络，请将其添加为该规则的来源。仅允许到受信任的来源的入站流量。

- f. 选择保存。
3. 在 Outbound (出站) 选项卡上选择 Edit (编辑)，然后删除自动创建的规则以允许到所有 IP 地址的出站流量。
4. 选择添加规则。
5. 对于 Type (类型)，请选择 All traffic (所有流量)。对于 Destination (目标)，将 Custom (自定义) 保持选中状态，然后从列表中选择 EC2 实例的安全组。这允许从应用程序负载均衡器和堡垒主机到 Ethereum 网络中的 EC2 实例的出站连接。
6. 选择添加规则。
7. 对于 Type (类型)，请选择 All traffic (所有流量)。对于 Destination (目标)，将 Custom (自定义) 保持选中状态，然后从列表中选择您当前编辑的安全组，例如 EthereumALB-SG。这允许应用程序负载均衡器与自身和堡垒主机进行通信。
8. 选择保存。

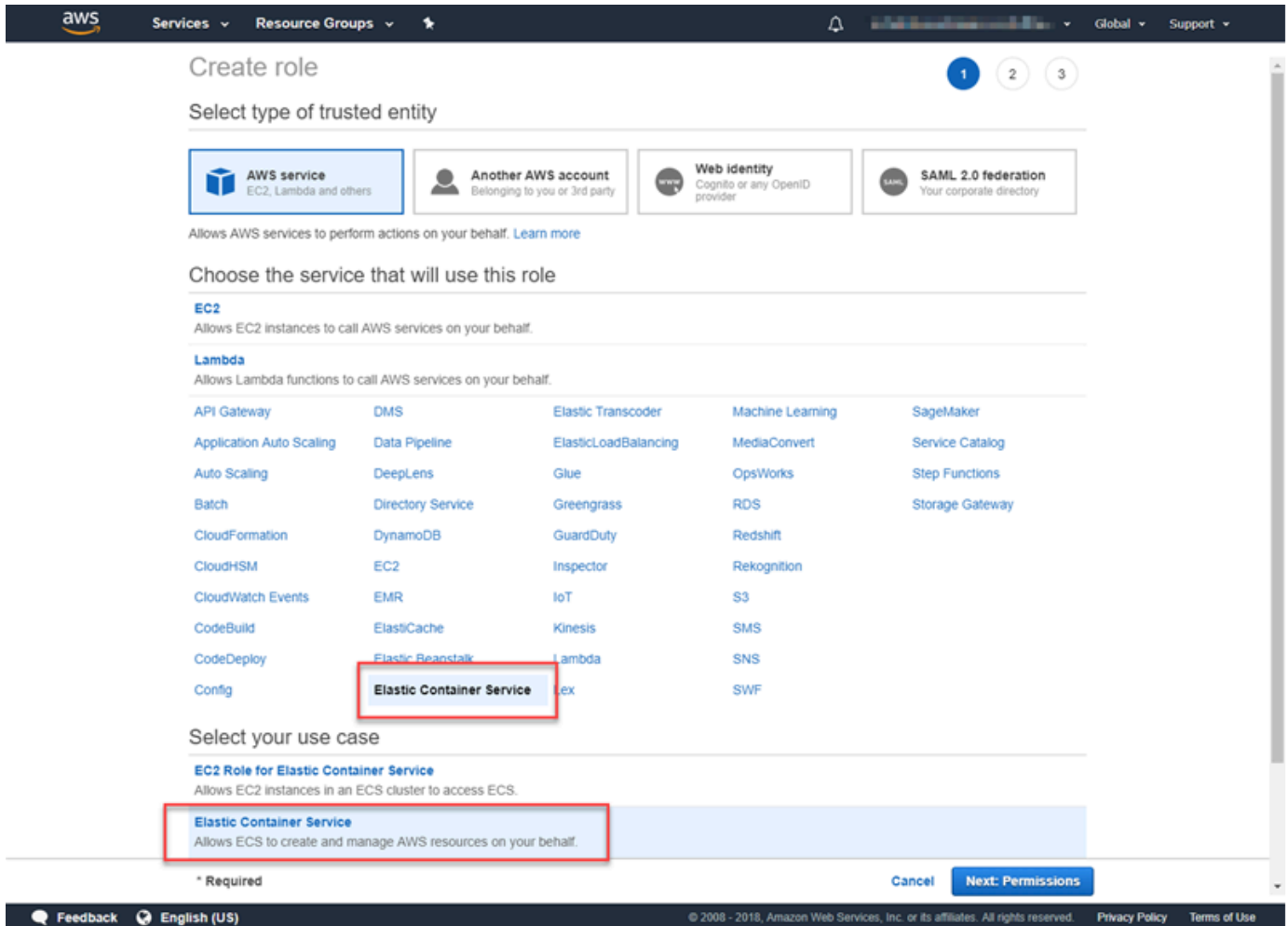
## 创建适用于 Amazon ECS 的 IAM 角色和 EC2 实例配置文件

当您使用此模板时，需要为 Amazon ECS 指定 IAM 角色，并指定 EC2 实例配置文件。附加到这些角色的权限策略允许您集群中的 AWS 资源和实例与其他 AWS 资源进行交互。有关更多信息，请参阅《IAM 用户指南》中的 [IAM 角色](#)。您可以使用 IAM 控制台 (<https://console.aws.amazon.com/iam/>) 为 Amazon ECS 和 EC2 实例配置文件设置 IAM 角色。

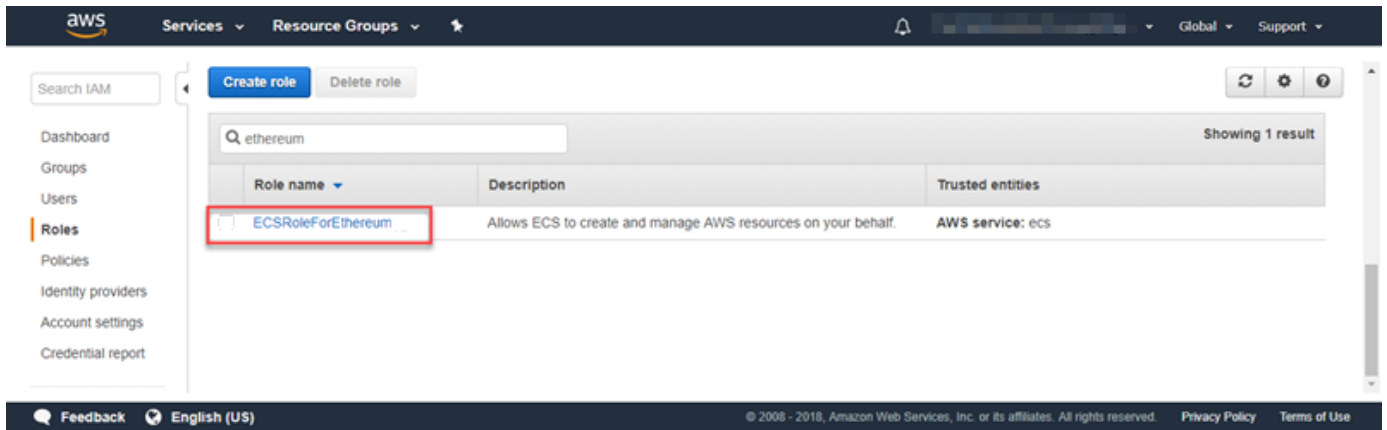
创建适用于 Amazon ECS 的 IAM 角色。

1. 使用 <https://console.aws.amazon.com/iam/> 打开 IAM 控制台。
2. 在导航窗格中，选择 **角色** 和 **创建角色**。
3. 在 Select type of trusted entity (选择受信任实体的类型) 下，选择 Amazon Web Services service (亚马逊云科技服务)。

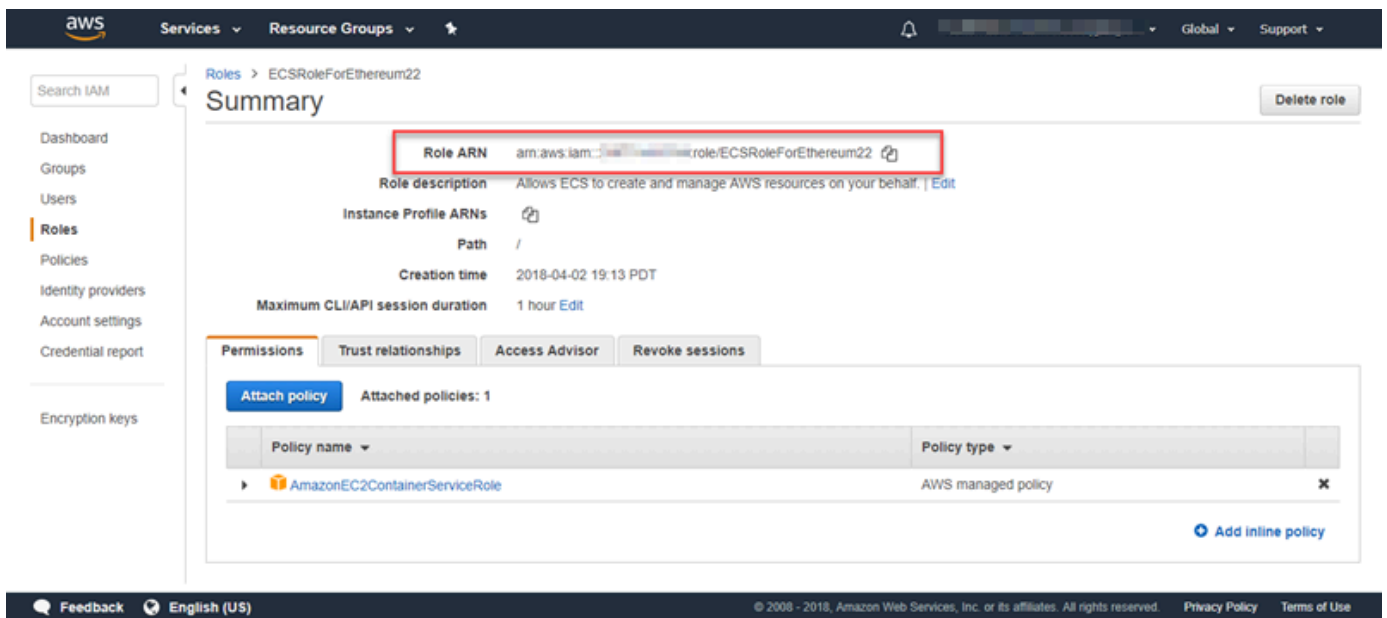
- 对于 Choose the service that will use this role (选择将使用此角色的服务)，选择 Elastic Container Service。
- 在 Select your use case (选择您的使用案例) 下，选择 Elastic Container Service (弹性容器服务) 和 Next:Permissions (下一步: 权限)。



- 对于权限策略，保留默认策略 (Amazon EC2 ContainerServiceRole) 处于选中状态，然后选择“下一步：查看”。
- 在角色名称中，输入一个可以帮助您识别角色的值，例如ECSRoleForEthereum。对于 Role Description (角色描述)，输入简短摘要。请记住角色名称，以备后用。
- 选择创建角色。
- 从列表中选择您刚刚创建的角色。如果您的账户中包含许多角色，则可搜索角色名称。



10. 复制 Role ARN (角色 ARN) 值并保存，以便再次复制。创建 Ethereum 网络时需要此 ARN。



您在模板中指定的 EC2 实例配置文件由以太坊网络中的 EC2 实例使用，用于与其他 AWS 服务进行交互。您可以为角色创建权限策略，创建角色 (自动创建同名实例配置文件)，然后将权限策略附加到角色。

### 创建 EC2 实例配置文件

1. 在导航窗格中，选择 Policies、Create policy。
2. 选择 JSON，并将默认策略语句替换为以下 JSON 策略：

```
{
  "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. 选择查看策略。
4. 例如，在“名称”中，输入一个可以帮助您识别此权限策略的值EthereumPolicyForEC2。对于Description (描述)，输入简短摘要。选择创建策略。

**Create policy** 1 2

**Review policy**

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

**Description**   
Maximum 1000 characters. Use alphanumeric and '+, @, \_' characters.

**Summary**

Service	Access level	Resource	Request condition
Allow (4 of 134 services) <a href="#">Show remaining 130</a>			
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**

- 依次选择角色和创建角色。
- 选择 EC2，然后选择 Next: Permissions (下一步: 权限)。
- 例如，在搜索字段中，输入您之前创建的权限策略的名称EthereumPolicyForEC2。
- 选中您以前创建的策略的复选标记，然后选择 Next: Review(下一步: 检查)。

**Create role** 1 2 3

**Attach permissions policies**

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

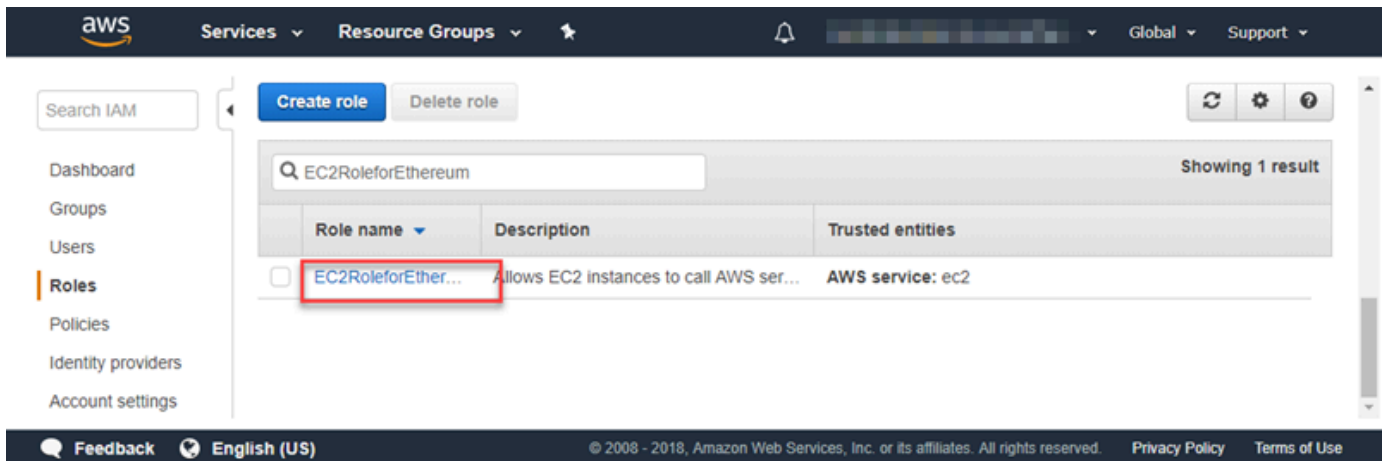
Filter: Policy type  Showing 1 result

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

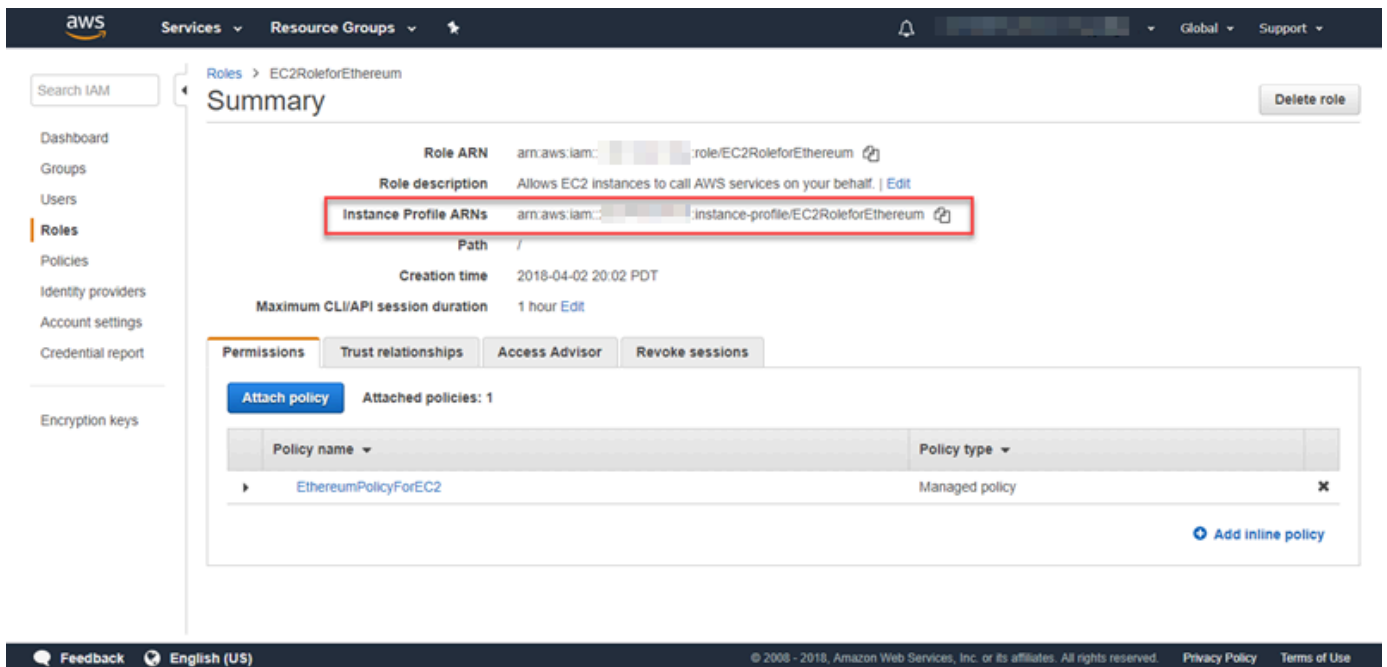
\* Required Cancel Previous **Next: Review**

- 例如，在角色名称中，输入一个可以帮助您识别角色的值EC2RoleForEthereum。对于 Role description (角色描述)，输入简短摘要。选择 Create role (创建角色)。

10. 从列表中选择您刚刚创建的角色。如果您的账户具有很多角色，您可以在 Search (搜索) 字段中输入角色名称。



11. 复制并保存 Instance Profile ARN (实例配置文件 ARN) 值，以便您可以再次复制该值。创建 Ethereum 网络时需要此 ARN。



## 创建堡垒主机

在本教程中，您创建一个堡垒主机。这是一个 EC2 实例，您可以使用该实例连接到 Ethereum 网络中的 Web 接口和实例。它的唯一用途是从 VPC 外部的受信任客户端转发 SSH 流量，以便它们可以访问 Ethereum 网络资源。

您可以设置堡垒主机，因为模板创建的应用程序负载均衡器是内部的，这意味着它仅路由内部 IP 地址。堡垒主机：

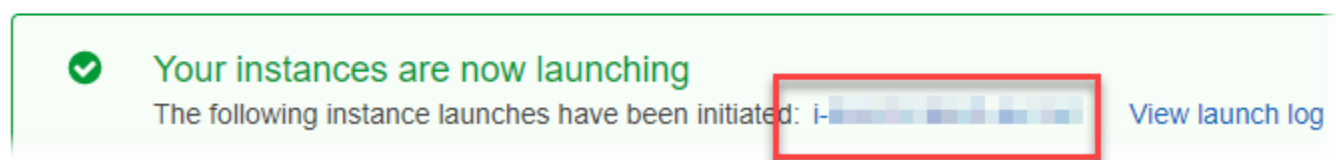
- 具有应用程序负载均衡器可识别的内部 IP 地址，因为您在以前创建的第二个公有子网中启动它。
- 具有子网分配的公有 IP 地址，VPC 外部的受信任来源可以访问该地址。
- 与您以前创建的应用程序负载均衡器的安全组关联，该安全组具有一个入站规则以允许来自受信任的客户端的 SSH 流量（端口 22）。

要能够访问 Ethereum 网络，需要设置受信任的客户端以通过堡垒主机进行连接。有关更多信息，请参阅 [连接 EthStats 并 EthExplorer 使用堡垒主机](#)。堡垒主机只是一种方法。您可以使用从受信任的客户端访问 VPC 中的私有资源的任何方法。

### 创建堡垒主机

1. 按照 Amazon EC2 用户指南中的前五个步骤[启动实例](#)。
2. 选择 Edit Instance Details。对于 Network (网络)，选择您以前创建的 VPC；对于 Subnet (子网)，选择您以前创建的第二个公有子网。将所有其他设置保留默认值。
3. 在出现提示时确认更改，然后选择 Review and Launch (检查并启动)。
4. 选择 Edit Security Groups (编辑安全组)。对于分配安全组，选择选择现有安全组。
5. 从安全组列表中，为您以前创建的应用程序负载均衡器选择安全组，然后选择 Review and Launch (检查并启动)。
6. 选择启动。
7. 记下实例 ID。稍后在[连接 EthStats 并 EthExplorer 使用堡垒主机](#)时，您需要使用该 ID。

## Launch Status



## 创建 Ethereum 网络

您使用本主题中的模板指定的以太坊网络会启动一个 CloudFormation 堆栈，该堆栈用于为以太坊网络创建由 EC2 实例组成的 Amazon ECS 集群。该模板依赖于您以前在[设置先决条件](#)中创建的资源。

当您使用模板启动 CloudFormation 堆栈时，它会为某些任务创建嵌套堆栈。完成后，您可以通过堡垒主机连接到网络的应用程序负载均衡器提供的资源，以验证 Ethereum 网络是否正在运行并且可访问。

使用适用于 Ethereum 的 Amazon Web Services Blockchain Templates 创建 Ethereum 网络

1. 请参阅 [AWS Blockchain Templates 入门](#)，并使用您的 AWS 区域的快速链接在 CloudFormation 控制台中打开适用于以太坊的最新 AWS 区块链模板。
2. 根据以下准则输入值：
  - 对于 Stack name (堆栈名称)，输入一个易于识别的名称。此名称用于堆栈创建的资源名称中。
  - 在 Ethereum Network Parameters (Ethereum 网络参数) 和 Private Ethereum Network Parameters (私有 Ethereum 网络参数) 下面，保留默认设置。

#### Warning

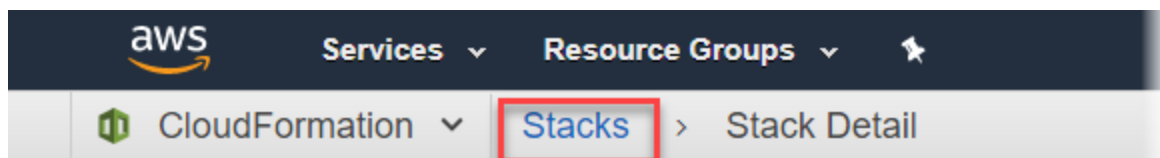
仅出于测试目的使用默认账户和关联的助记词。请勿使用默认账户集发送真实的 Ether，因为任何有权访问助记词的人都可以从账户中访问或窃取 Ether。相反，应出于生产目的指定自定义账户。与默认账户关联的助记词为 outdoor father modify clever trophy abandon vital feel portion grit evolve twist。

- 在平台配置下面，保留默认设置，这会创建一个包含 EC2 实例的 集群。另一选项是 docker-local，它使用单个 EC2 实例创建 Ethereum 网络。
- 在 EC2 configuration (EC2 配置) 下面，根据以下准则选择选项：
  - 对于 EC2 Key Pair (EC2 密钥对)，选择一个密钥对。有关创建密钥对的信息，请参阅 [创建密钥对](#)。
  - 对于 EC2 Security Group (EC2 安全组)，选择您以前在 [创建安全组](#) 中创建的安全组。
  - 对于 EC2 Instance Profile ARN (EC2 实例配置文件 ARN)，输入您以前在 [创建适用于 Amazon ECS 的 IAM 角色和 EC2 实例配置文件](#) 中创建的实例配置文件的 ARN。
- 在 VPC network configuration (VPC 网络配置) 下面，根据以下准则选择选项：
  - 对于 VPC ID，选择您以前在 [创建 VPC 和子网](#) 中创建的 VPC。
  - 对于以太坊网络子网 IDs，请选择您在前面步骤中创建的单个私有子网 [To create the VPC](#)。
- 在 ECS cluster configuration (ECS 集群配置) 下面，保留默认值。这会创建包含三个 EC2 实例的 ECS 集群。
- 在 Application Load Balancer configuration (ECS only) (应用程序负载均衡器配置 (仅 ECS)) 下面，根据以下准则选择选项：

- 对于 Application Load Balancer 子网，请从您之前提到的[list of subnets](#)中选择两个公有子网。
  - 对于 Application Load Balancer Security Group (应用程序负载均衡器安全组)，选择您以前在[创建安全组](#)中创建的应用程序负载均衡器的安全组。
  - 对于 IAM 角色，输入您以前在[创建适用于 Amazon ECS 的 IAM 角色和 EC2 实例配置文件中](#)创建的 ECS 角色的 ARN。
3. 在下方 EthStats，根据以下准则选择选项：
- 对于 Deploy EthStats，保留默认设置，该设置为 true。
  - 对于“EthStats 连接密钥”，键入一个至少为六个字符的任意值。
- 在“部署”下 EthExplorer，保留“部署”的默认设置 EthExplorer，该设置为 true。
- 在 Other parameters (其他参数) 下面，保留 Nested Template S3 URL Prefix (嵌套模板 S3 URL 前缀) 的默认值并记下该值。您可以在此处找到嵌套模板。
3. 将所有其他设置保留为默认值，选中确认复选框，然后选择 Create (创建)。

将显示 CloudFormation 启动的根堆栈的堆栈详细信息页面。

4. 要监控根堆栈和嵌套堆栈的进度，请选择 Stacks (堆栈)。



## MyFirstEthereumStack

Stack name: MyFirstEthereumStack

5. 当所有堆栈的状态显示CREATE\_COMPLETE时，您可以连接到 Ethereum 用户界面，以验证网络是否正在运行且可以访问。当您使用 ECS 容器平台时，URLs 根堆栈的“输出”选项卡上可以通过 Application Load Balancer 连接到 EthExplorer、和 EthJson RPC。EthStats

### Important

在通过客户端计算机上的堡垒主机设置代理连接之前，您将无法直接连接到这些服务器 URLs 或直接连接 SSH。有关更多信息，请参阅 [连接 EthStats 并 EthExplorer 使用堡垒主机](#)。

The screenshot shows the AWS CloudFormation console. At the top, there are navigation tabs 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 and highlighted with a red box. Below the table, the 'Outputs' tab is active, showing a table of stack outputs. The 'EthStatsURL' output is highlighted with a red box, showing a URL that includes 'us-west-2.elb.amazonaws.com'. Other outputs include 'EthExplorerURL' and 'EthJsonRPCURL'.

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:8080	Visit this URL to view transactions on yo...	
EthJsonRPCURL	http://MyFir-...us-west-2.elb.amazonaws.com:8545	Use this URL to access the Geth JSON ...	

## 连接 EthStats 并 EthExplorer 使用堡垒主机

要连接到本教程中的 Ethereum 资源，您可以通过堡垒主机设置 SSH 端口转发（SSH 隧道）。以下说明演示了如何执行此操作，以便您可以连接 EthStats 和 EthExplorer URLs 使用浏览器。在下面的说明中，您先在本地端口上设置 SOCKS 代理。然后，您可以使用浏览器扩展程序将此转发端口用于您的以太坊网络 URLs。 [FoxyProxy](#)

如果使用 Mac OS 或 Linux，请使用 SSH 客户端设置到堡垒主机的 SOCKS 代理连接。如果您是 Windows 用户，请使用 PuTTY。在连接之前，请确认在您以前设置的应用程序负载均衡器的安全组中将您使用的客户端计算机指定为入站 SSH 流量的允许来源。

使用 SSH 连接到具有 SSH 端口转发的堡垒主机

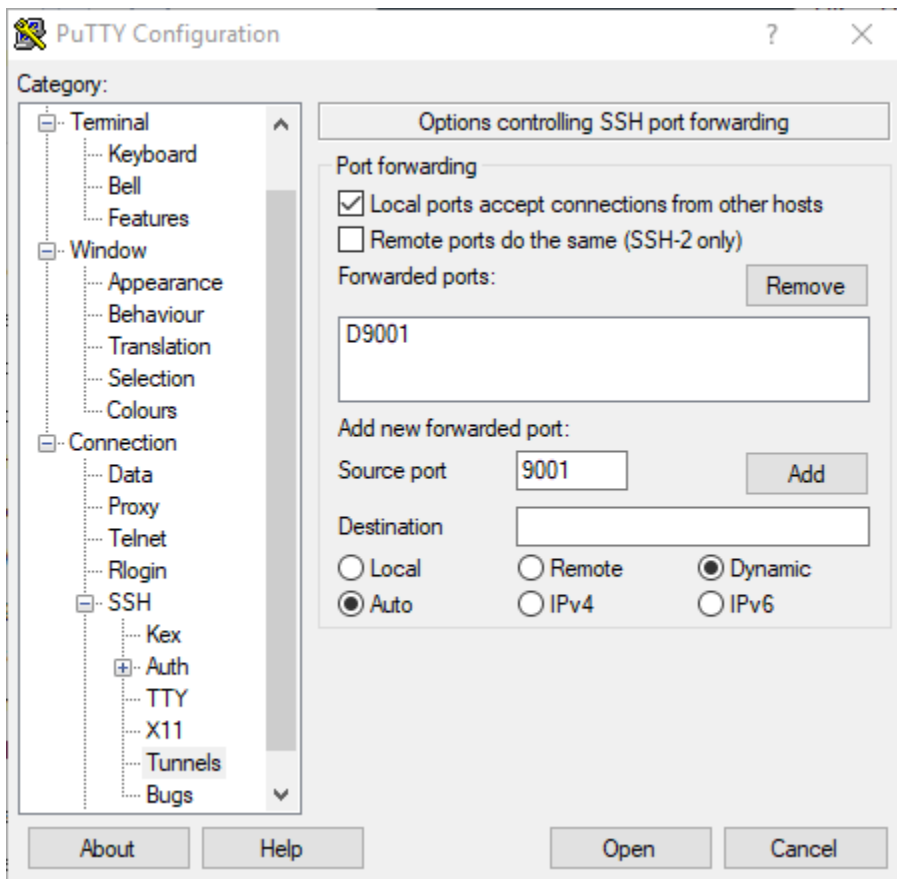
- 按照 Amazon EC2 用户指南中[使用 SSH 连接您的 Linux 实例](#)中的步骤进行操作。对于[连接到 Linux 实例](#)过程的步骤 4，将添加到 SSH 命令中，指定 Ethereum 的 AWS Blockchain Template 配置中指定的相同密钥对，然后指定堡垒主机的 DNS 名称。

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

使用 PuTTY 连接到具有 SSH 端口转发的堡垒主机 (Windows)

1. [使用您在以太坊的 AWS 区块链模板配置中指定的密钥对，按照 Amazon EC2 用户指南中使用 PuTTY 从 Windows 连接到 Linux 实例中的步骤，完成启动 PuTTY 会话过程的第 7 步。](#)
2. 在 PuTTY 中，在 Category (类别) 下面选择 Connection (连接)、SSH 和 Tunnels (隧道)。
3. 对于 Port forwarding (端口转发)，选择 Local ports accept connections from other hosts (本地端口接受来自其他主机的连接)。
4. 在 Add new forwarded port (添加新的转发端口) 下面：
  - a. 对于 Source port (源端口)，输入 9001。这是我们选择的任意未使用的端口，如有必要，您可以选择其他端口。
  - b. 将 Destination (目标) 保留空白。
  - c. 选择 Dynamic (动态)。
  - d. 选择添加。

对于 Forwarded ports (转发的端口)，D9001 应如下所示。



5. 选择 Open (打开)，然后根据密钥配置要求在堡垒主机中进行身份验证。将连接保持打开状态。

打开 Putty 连接后，您现在将系统或浏览器扩展程序配置为使用以太坊网络的转发端口。URLs 以下说明基于使用 FoxyProxy 标准转发连接，基于和和的 URL 模式 EthStats EthExplorer 和端口 9001 (您之前将其设置为转发端口)，但您可以使用任何您喜欢的方法。

### 配置为使用 FoxyProxy 以太坊网络的 SSH 隧道 URLs

该过程是根据 Chrome 编写的。如果您使用其他浏览器，请将设置和顺序转换为该浏览 FoxyProxy 器的版本。

1. 下载并安装 FoxyProxy 标准浏览器扩展程序，然后根据浏览器的说明打开选项。
2. 选择 Add New Proxy (添加新代理)。
3. 在 General (常规) 选项卡上，确保 Enabled (已启用) 代理，并输入 Proxy Name (代理名称) 和 Proxy Notes (代理注释) 以帮助您标识该代理配置。
4. 在 Proxy Details (代理详细信息) 选项卡上，选择 Manual Proxy Configuration (手动代理配置)。对于 Host or IP Address (主机或 IP 地址) (在某些版本中为 Server or IP Address (服务器或 IP 地址))，输入 localhost。对于 Port (端口)，输入 9001。选择 SOCKS Proxy? (SOCKS 代理?)。

5. 在 URL Pattern (URL 模式) 选项卡上，选择 Add New Pattern (添加新模式)。
6. 在“模式名称”中，输入一个易于识别的名称；对于 URL 模式，输入与 URLs 您使用模板创建的所有以太坊资源相匹配的模式，例如 `http://internal-MyUser-loadb-*`。有关查看的信息 URLs，请参阅 [Ethereum URLs](#)。
7. 将其他设置保留默认选项，然后选择 Save (保存)。

现在，您可以使用使用模板创建的根堆栈的 Outputs 选项卡连接到以太坊 URLs，该选项卡可在 CloudFormation 控制台上使用。

## 清理资源

CloudFormation 便于清理堆栈创建的资源。删除堆栈后，堆栈创建的所有资源都会被删除。

### 删除模板创建的资源

- 打开 CloudFormation 控制台，选择之前创建的根堆栈，选择操作，删除。

您以前创建的根堆栈和关联的嵌套堆栈的 Status (状态) 更新为 `DELETE_IN_PROGRESS`。

可以选择删除您为 Ethereum 网络创建的先决条件。

### 删除 VPC

- 打开 Amazon VPC 控制台，选择您之前创建的 VPC，然后选择操作、删除 VPC。这也将删除与该 VPC 关联的子网、安全组和 NAT 网关。

### 删除 IAM 角色和 EC2 实例配置文件

- 打开 IAM 控制台，选择角色。选择您以前创建的 ECS 和 EC2 的角色，然后选择 Delete (删除)。

### 终止堡垒主机的 EC2 实例

- 打开 Amazon EC2 控制面板，选择运行实例，选择您未堡垒主机创建的 EC2 实例，选择操作、实例状态、终止。

# Amazon Web Services Blockchain Templates 和特征

本节提供的链接可供您立即开始创建区块链网络，还提供配置选项信息，以及在 AWS 上设置该网络的先决条件。

可用模板如下：

- [适用于 Ethereum 的 Amazon Web Services Blockchain Templates](#)
- [适用于 Hyperledger Fabric 的 Amazon Web Services Blockchain Templates](#)

AWS Blockchain Templates 在以下区域中可用：

- 美国西部 ( 俄勒冈州 ) 区域 ( us-west-2 )
- 美国东部 ( 弗吉尼亚北部 ) 区域 (us-east-1)
- 美国东部 ( 俄亥俄 ) 区域 (us-east-2)

## Note

在上述未列区域运行模板会在美国东部 ( 弗吉尼亚州北部 ) 区域 (us-east-1) 中启动资源。

## 使用适用于 Ethereum 的 Amazon Web Services Blockchain Templates

Ethereum 是一个区块链框架，它使用 Ethereum 特定的语言 Solidity 来运行智能合同。Homestead 是 Ethereum 的最新版本。有关更多信息，请参阅 [Ethereum Homestead 文档](#) 和 [Solidity 文档](#)。

### 启动链接

有关使用以太坊 [模板 CloudFormation 在特定区域启动的链接](#)，请参阅 [AWS Blockchain 模板入门](#)。

### Ethereum 选项

如果您使用模板配置 Ethereum 网络，您做出的选择将决定后续要求：

- [选择容器平台](#)

- [选择私有或公有 Ethereum 网络](#)
- [更改默认账户和助记词](#)

## 选择容器平台

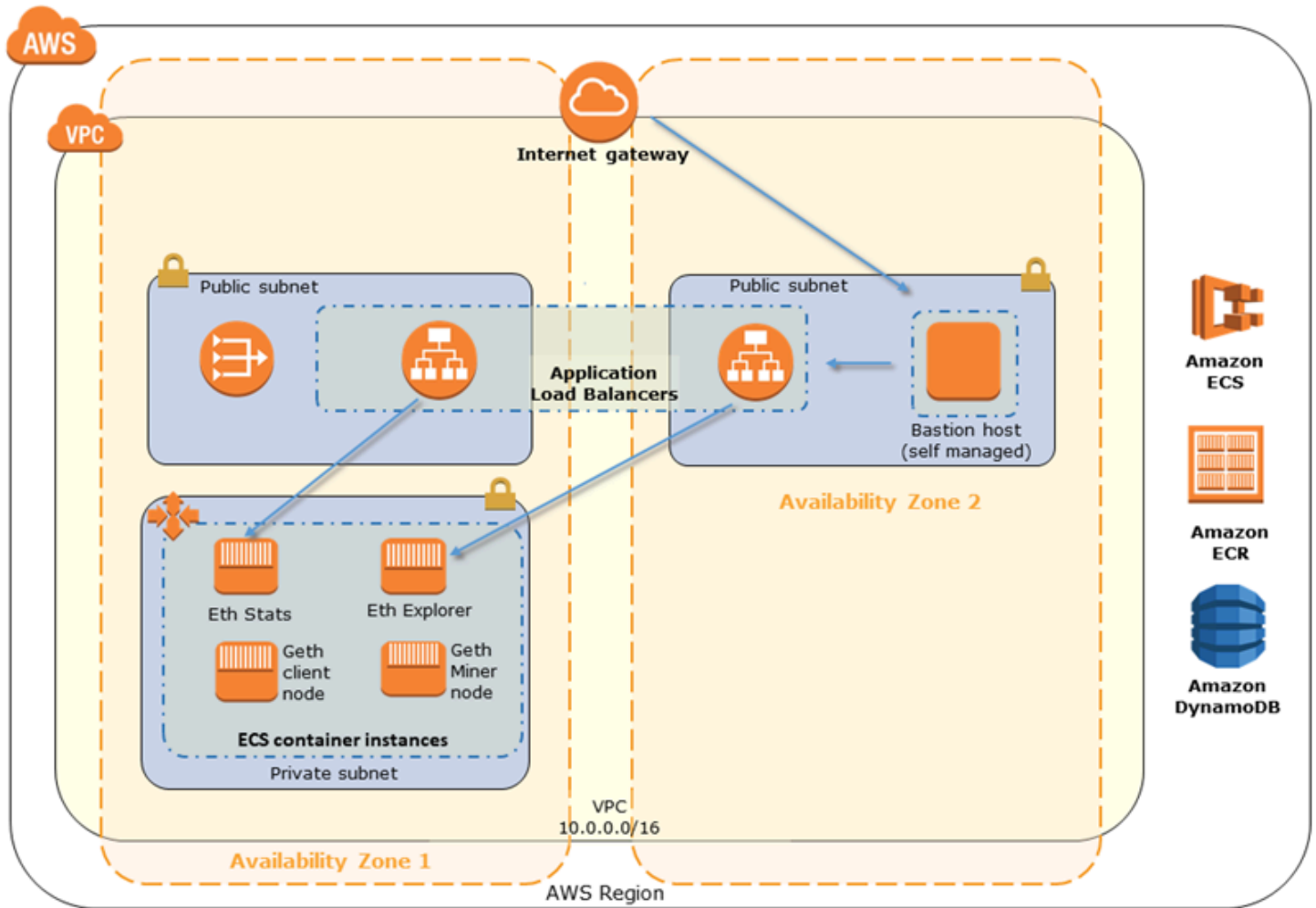
AWS Blockchain Templates 使用 Amazon ECR 中存储的 Docker 容器部署区块链软件。AWS Blockchain Templates 提供两种备选容器平台：

- ecs —指定 Ethereum 在 Amazon EC2 实例的 Amazon ECS 集群上运行。
- docker-local—指定 Ethereum 在单个 EC2 实例上运行。

### 使用 Amazon ECS 容器平台

通过使用 Amazon ECS，您可以在包含多个 EC2 实例的 ECS 集群上创建 Ethereum 网络，并具有应用程序负载均衡器和相关的资源。有关使用 Amazon ECS 配置的更多信息，请参阅 [Amazon Web Services Blockchain Templates 入门](#) 教程。

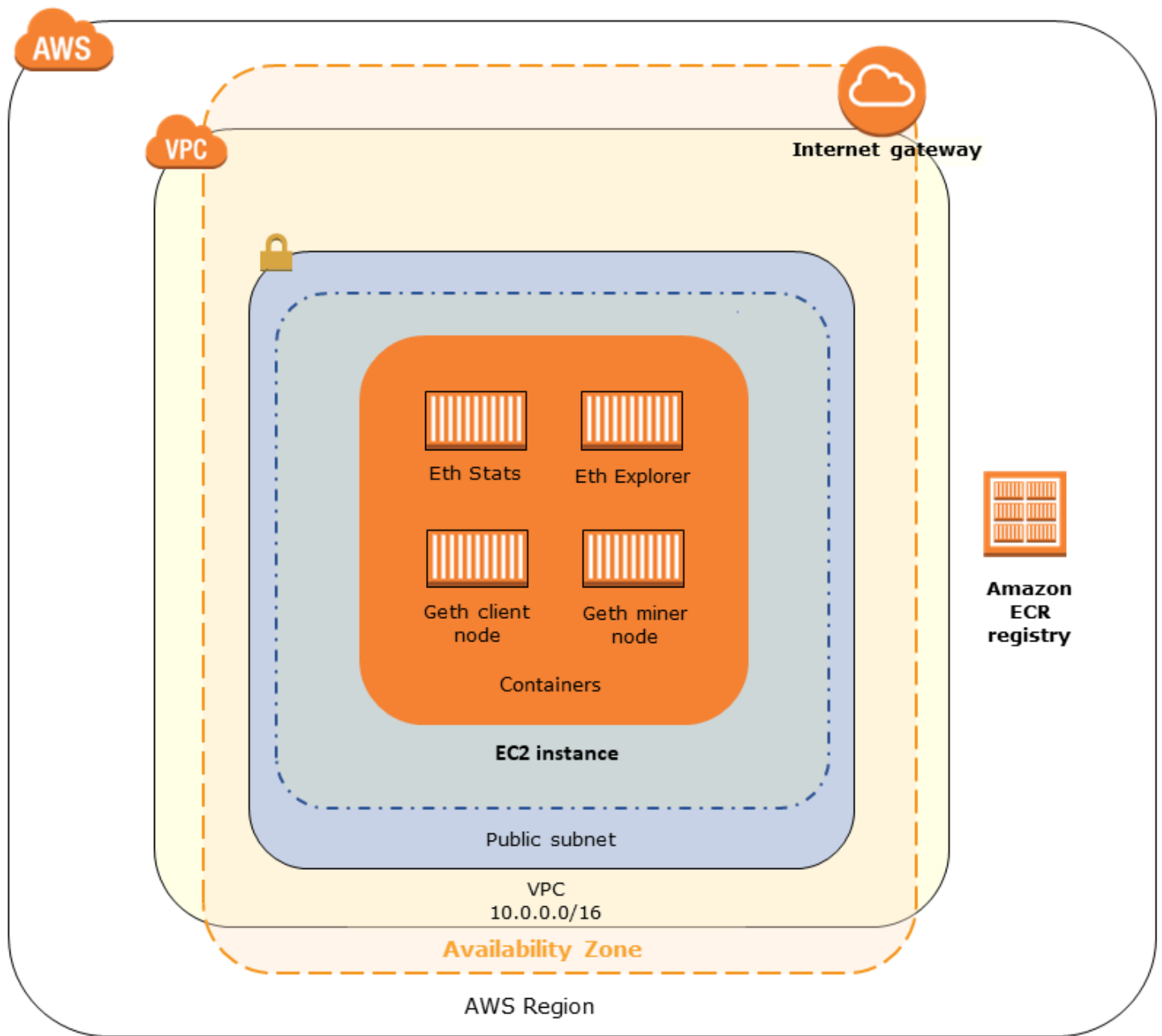
下图描述了使用 ECS 容器平台选项通过模板创建的 Ethereum 网络：



## 使用 Docker-Local 平台

您也可以在单独的 Amazon EC2 实例中启动 Ethereum 容器。所有容器都在单个 EC2 实例上运行。这是简化的设置。

下图描述了使用 docker-local 容器平台选项通过模板创建的 Ethereum 网络：



## 选择私有或公有 Ethereum 网络

选择 1–4 以外的 Ethereum Network ID (Ethereum 网络 ID) 值，可在您定义的网络中，使用您指定的私有网络参数创建私有 Ethereum 节点。

如果您选择 1-4 之间的值作为 Ethereum Network ID (Ethereum 网络 ID)，您创建的 Ethereum 节点会加入公有 Ethereum 网络。您可以忽略私有网络设置及其默认值。如果您选择将 Ethereum 节点加入公有 Ethereum 网络，请确保您的网络中的相应服务可通过 Internet 访问。

## 更改默认账户和助记词

助记词是一组随机的单词，可用于为任何网络上的关联账户生成以太坊钱包（即 private/public 密钥对）。助记词可用于访问关联账户的 Ether。我们创建了一个默认助记符，它与 Ethereum 模板使用的默认账户相关联。

### Warning

仅出于测试目的使用默认账户和关联的助记词。请勿使用默认账户集发送真实的 Ether，因为任何有权访问助记词的人都可以从账户中访问或窃取 Ether。相反，应出于生产目的指定自定义账户。与默认账户关联的助记词为 outdoor father modify clever trophy abandon vital feel portion grit evolve twist。

## 先决条件

在使用 AWS Blockchain Template 设置 Ethereum 网络时，必须满足下面列出的最低要求。该模板需要以下每个类别列出的 AWS 组件：

### 主题

- [访问 Ethereum 资源的先决条件](#)
- [IAM 的先决条件](#)
- [安全组先决条件](#)
- [VPC 先决条件](#)
- [EC2 实例配置文件和 ECS 角色的示例 IAM 权限](#)

### 访问 Ethereum 资源的先决条件

先决条件	对于 ECS 平台	对于 Docker-Local
用于访问 EC2 实例的 Amazon EC2 密钥对。该密钥必须与 ECS 集群和其他资源同在一个区域中。	✓	✓
具有内部地址的面向 Internet 的组件（例如堡垒主机或面向	✓	✓（具有私有子网）

先决条件	对于 ECS 平台	对于 Docker-Local
Internet 的负载均衡器 )，允许流量从该地址进入应用程序负载均衡器。ECS 平台需要使用该组件，因为模板创建内部负载均衡器以保护安全。在 EC2 实例位于私有子网时 ( 建议的配置 )，docker-local 平台需要使用该组件。有关配置堡垒主机的信息，请参阅 <a href="#">创建堡垒主机</a> 。		

## IAM 的先决条件

先决条件	对于 ECS 平台	对于 Docker-Local
有权使用所有相关服务的 IAM 主体 ( 用户或组 )。	✓	✓
Amazon EC2 实例配置文件，其中的权限允许 EC2 实例与其他服务进行交互。有关更多信息，请参阅 <a href="#">To create an EC2 instance profile</a> 。	✓	✓
IAM 角色，其权限允许 Amazon ECS 与其他服务进行交互。有关更多信息，请参阅 <a href="#">创建 ECS 角色和权限</a> 。	✓	

## 安全组先决条件

先决条件	对于 ECS 平台	对于 Docker-Local
EC2 实例的安全组需满足以下要求：	✓	✓
<ul style="list-style-type: none"> <li>出站规则，允许流量流至 0.0.0.0/0 (默认)。</li> </ul>	✓	✓
<ul style="list-style-type: none"> <li>入站规则，它允许来自自身 (同一安全组) 的流量。</li> </ul>	✓	✓
<ul style="list-style-type: none"> <li>入站规则，它允许来自应用程序负载均衡器的安全组的所有流量。</li> </ul>	✓	
<ul style="list-style-type: none"> <li>允许来自可信外部来源 (例如客户端计算机的 IP CIDR) 的 HTTP EthStats (端口 80)、(在端口 8080 上提供服务)、通过 HTTP 的 JSON RPC (端口 8545) 和 SSH (端口 22) 的入站规则。</li> </ul>		✓
<p>应用程序负载均衡器的安全组需满足以下要求：</p> <ul style="list-style-type: none"> <li>入站规则，它允许来自自身 (同一安全组) 的流量。</li> <li>入站规则，允许来自 EC2 实例的安全组的全部流量。</li> <li>出站规则，它仅允许进入 EC2 实例的安全组的所有流量。有关更多信息，请参阅 <a href="#">创建安全组</a>。</li> </ul>	✓	

先决条件	对于 ECS 平台	对于 Docker-Local
<ul style="list-style-type: none"> <li>如果将该相同安全组与堡垒主机关联，则为允许来自受信任来源的 SSH ( 端口 22 ) 流量的进站规则。</li> <li>如果堡垒主机或其他面向 Internet 的组件位于不同的安全组中，则为允许来自该组件的流量的进站规则。</li> </ul>		

## VPC 先决条件

先决条件	对于 ECS 平台	对于 Docker-Local
弹性 IP 地址，用于访问 Ethereum 服务。	✓	✓
运行 EC2 实例的子网。我们强烈建议使用私有子网。	✓	✓
两个可公开访问的子网。每个子网必须位于彼此不同的可用区中，并且一个子网位于与 EC2 实例的子网相同的可用区中。	✓	

## EC2 实例配置文件和 ECS 角色的示例 IAM 权限

您在使用模板时需指定 EC2 实例配置文件 ARN，作为参数之一。如果您使用 ECS 容器平台，还需指定 ECS 角色 ARN。附加到这些角色的权限策略允许集群中的 AWS 资源和实例与其他 AWS 资源进行交互。有关更多信息，请参阅《IAM 用户指南》中的 [IAM 角色](#)。使用以下策略语句和过程作为创建权限的起点。

## EC2 实例配置文件的示例权限策略

以下权限策略说明了选择 ECS 容器平台的情况下 EC2 实例配置文件允许的操作。Docker-Local 容器平台中也可以使用相同的策略语句，只需移除 ecs 上下文密钥以限制访问。

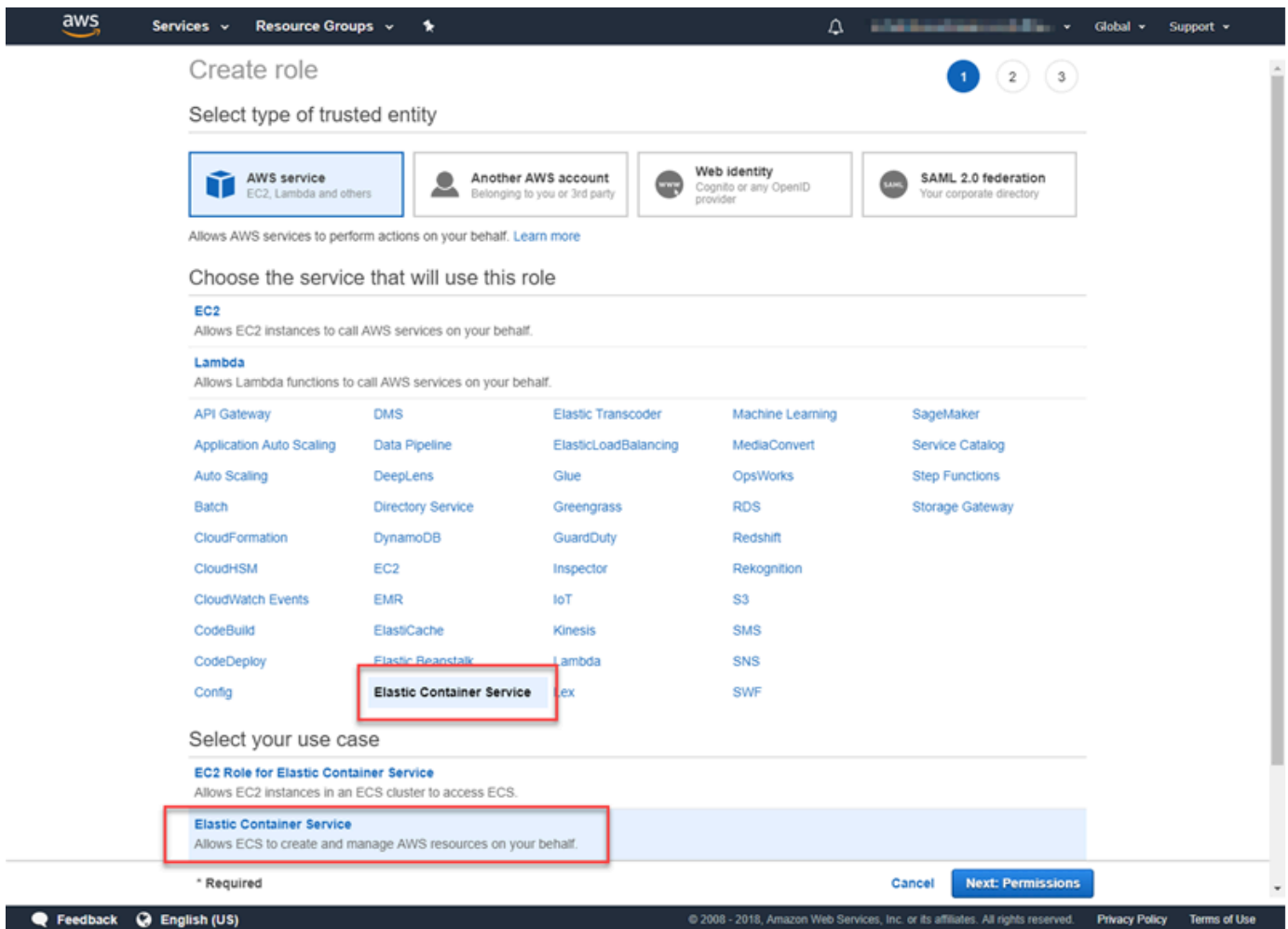
```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ecs:CreateCluster",
        "ecs:DeregisterContainerInstance",
        "ecs:DiscoverPollEndpoint",
        "ecs:Poll",
        "ecs:RegisterContainerInstance",
        "ecs:StartTelemetrySession",
        "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": "*"
    }
  ]
}
```

### 创建 ECS 角色和权限

对于附加到 ECS 角色的权限，我们建议您从 Amazon EC2 ContainerServiceRole 权限策略开始。使用以下过程创建角色并附加此权限策略。使用 IAM 控制台查看此策略中最多的 up-to-date 权限。

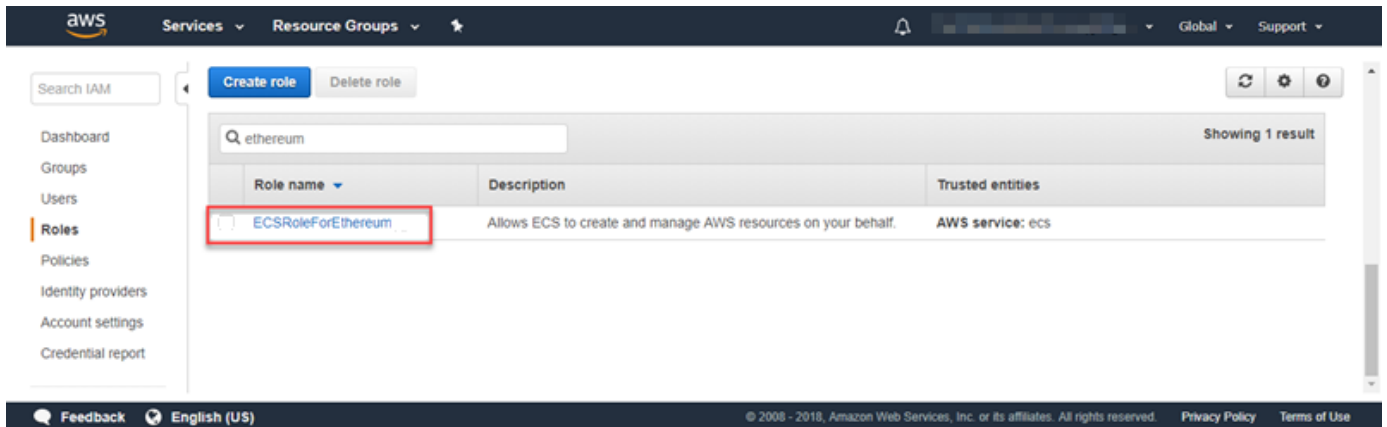
创建适用于 Amazon ECS 的 IAM 角色。

1. 使用 <https://console.aws.amazon.com/iam/> 打开 IAM 控制台。
2. 在导航窗格中，选择 **角色** 和 **创建角色**。
3. 在 **Select type of trusted entity** (选择受信任实体的类型) 下，选择 **Amazon Web Services service** (亚马逊云科技服务)。
4. 对于 **Choose the service that will use this role** (选择将使用此角色的服务)，选择 **Elastic Container Service**。
5. 在 **Select your use case** (选择您的使用案例) 下，选择 **Elastic Container Service** (弹性容器服务) 和 **Next:Permissions** (下一步: 权限)。

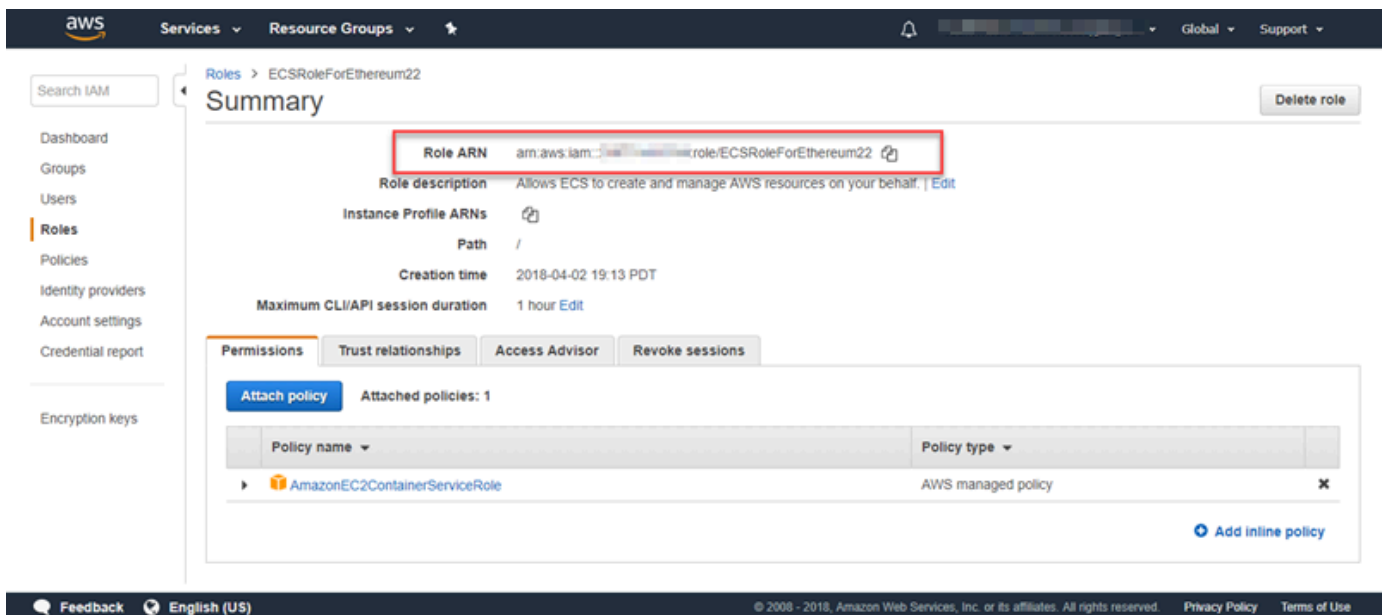


6. 对于权限策略，保留默认策略 (Amazon EC2 ContainerServiceRole) 处于选中状态，然后选择“下一步：查看”。
7. 在角色名称中，输入一个可以帮助您识别角色的值，例如ECSRoleForEthereum。对于 Role Description (角色描述)，输入简短摘要。请记住角色名称，以备后用。

- 选择创建角色。
- 从列表中选择您刚刚创建的角色。如果您的账户中包含许多角色，则可搜索角色名称。



- 复制 Role ARN (角色 ARN) 值并保存，以便再次复制。创建 Ethereum 网络时需要此 ARN。



## 连接到 Ethereum 资源

使用模板创建的根堆栈显示 CREATE\_COMPLETE 后，您可以使用控制台连接到以太坊资源。CloudFormation 连接方式依赖于您所选的容器平台，即 ECS 或 Docker-Local：

- ECS — 根堆栈的输出选项卡为应用程序负载均衡器上运行的服务提供链接。出于安全考虑，URLs 这些无法直接访问。要进行连接，您可以设置并使用堡垒主机以作为它们的连接代理。有关更多信息，请参阅下面的 [使用堡垒主机的代理连接](#)。

- docker-local — 您使用托管 Ethereum 服务的 EC2 实例的 IP 地址进行连接，如下所示。使用 EC2 控制台查找模板创建 *ec2-IP-address* 的实例。
  - EthStats—使用 `http://ec2-IP-address`
  - EthExplorer—使用 `http://:8080 ec2-IP-address`
  - EthJsonRpc—使用 `http://:8545 ec2-IP-address`

如果您为 Ethereum Network Subnet ID (Ethereum 网络子网 ID) 指定了公有子网 (模板中的 List of VPC Subnets to use (要使用的 VPC 子网列表))，您可以直接进行连接。您的客户端必须是 SSH 的入站流量 (端口 22) 以及所列端口的可信来源。这是由您使用 Ethereum 的 AWS Blockchain Template 指定的 EC2 安全组决定的。

如果您指定了私有子网，您可以设置并使用堡垒主机以作为到这些地址的连接代理。有关更多信息，请参阅下面的 [使用堡垒主机的代理连接](#)。

## 使用堡垒主机的代理连接

包含某些配置的 Ethereum 服务可能不会公开。在这种情况下，您可以通过堡垒主机连接到 Ethereum 资源。有关堡垒主机的更多信息，请参阅 [Linux 堡垒主机快速入门指南中的 Linux 堡垒主机架构](#)。

堡垒主机是 EC2 实例。确保满足以下要求：

- 堡垒主机的 EC2 实例位于启用了自动分配公有 IP 且具有互联网网关的公有子网内。
- 堡垒主机拥有允许 ssh 连接的密钥对。
- 堡垒主机与允许来自连接的客户端的入站 SSH 流量的安全组关联。
- 分配给 Ethereum 主机的安全组 (例如，如果 ECS 是容器平台，则分配给应用程序负载均衡器；如果 docker-local 是容器平台，则分配给主机 EC2 实例) 允许 VPC 内的所有端口上的入站流量。

设置堡垒主机后，请确保连接的客户端使用堡垒主机作为代理。以下示例演示了如何使用 Mac OS 设置代理连接。*BastionIP* 替换为堡垒主机 EC2 实例的 IP 地址和 *MySshKey.pem* 您复制到堡垒主机的 key pair 文件。

在命令行键入以下内容：

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

这将为本地计算机上的端口 9001 设置至堡垒主机的端口转发。

接下来，将您的浏览器或系统设置为使用适用于localhost:9001 的 SOCKS 代理。例如，使用 MacOS，选择 System Preferences (系统首选项)、Network (网络)、Advanced (高级)，再选择 SOCKS proxy (SOCKS 代理)，然后键入 localhost:9001。

在 Chrome 中使用 FoxyProxy 标准版，选择“更多工具”、“扩展程序”。在“FoxyProxy 标准”下，选择“详细信息”、“扩展选项”、“添加新代理”。选择 Manual Proxy Configuration (手动代理配置)。对于 Host or IP Address (主机或 IP 地址)，键入 localhost，对于 Port (端口)，键入 9001。选择 SOCKS Proxy? (SOCKS 代理?)、Save (保存)。

您现在应能连接到模板输出中列出的 Ethereum 主机地址。

## 使用适用于 Hyperledger Fabric 的 Amazon Web Services Blockchain Templates

Hyperledger Fabric 是一个区块链框架，它运行使用 Go 编写的智能合同 (名为 chaincode)。您可以利用 Hyperledger Fabric 创建私有网络，限制可以连入并参与该网络的对等方。有关 Hyperledger Fabric 的更多信息，请参阅 [Hyperledger Fabric](#) 文档。有关 chaincode 的更多信息，请参阅 [Hyperledger Fabric](#) 中的 [开发者 Chaincode](#) 主题。

AWS Blockchain Template for Hyperledger Fabric 仅支持一个 docker-local 容器平台，这意味着 Hyperledger Fabric 容器部署在单个 EC2 实例上。

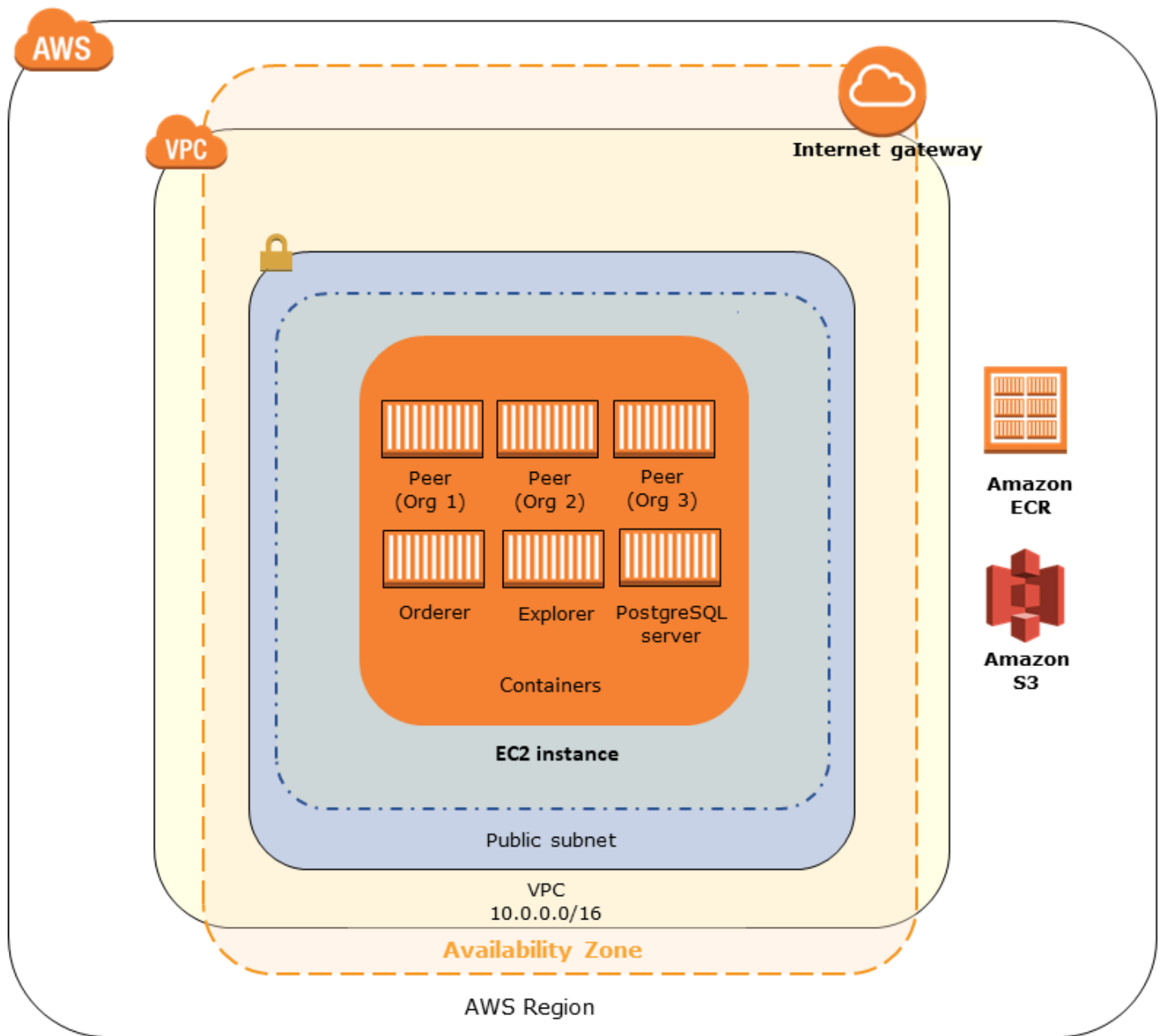
### 启动链接

有关 [使用 Hyperledger Fabric 模板 CloudFormation 在特定区域启动的链接](#)，请参阅 [AWS Blockchain 模板入门](#)。

## 适用于 Hyperledger Fabric 组件的 Amazon Web Services Blockchain Templates

AWS Blockchain Template for Hyperledger Fabric 使用 Docker 创建一个 EC2 实例，并在该实例上使用容器启动 Hyperledger Fabric 网络。该网络包括一个订单服务和三个组织，每个组织有一个对等服务。模板还会启动一个 Hyperledger Explorer 容器，用于浏览区块链数据。还会启动一个 PostgreSQL 服务器容器来支持 Hyperledger Explorer。

下图描述了使用模板创建的 Hyperledger Fabric 网络：



## 先决条件

使用模板启动 Hyperledger Fabric 网络之前，请确保满足以下要求：

- 您使用的 IAM 主体（用户或组），必须有权使用所有相关服务。
- 您必须有权使用密钥对来访问 EC2 实例（例如，使用 SSH）。密钥必须与实例存在于同一区域。

- 您必须具有附加了权限策略的 EC2 实例配置文件，从而允许访问 Amazon S3 和 Amazon Elastic Container Registry (Amazon ECR) 以拉取容器。有关权限策略的示例，请参阅 [EC2 实例配置文件的示例 & IAM; 权限](#)。
- 您必须拥有一个带有公有子网的 Amazon VPC 网络，或者一个带有 NAT 网关和弹性 IP 地址的私有子网，这样才能访问 Amazon S3 和 Amazon ECR。 CloudFormation
- 您必须具有一个包含入站规则的 EC2 安全组，以允许来自需要使用 SSH 连接到实例的 IP 地址的 SSH 流量（端口 22），对需要连接到 Hyperledger Explorer 的客户端（端口 8080）同样适用。

## EC2 实例配置文件的示例 & IAM; 权限

您在使用 AWS Blockchain Template for Hyperledger Fabric 模板时需指定 EC2 实例配置文件 ARN，作为参数之一。使用以下策略语句，作为附加到 EC2 角色和实例配置文件的权限策略的起点。

### JSON

```
{
  "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": "*"
    }
  ]
}
```

## 连接到 Hyperledger Fabric Resources

在使用模板创建的根堆栈显示CREATE\_COMPLETE后，您可以连接到 EC2 实例上的 Hyperledger Fabric 资源。如果您指定了公有子网，可以像连接任何其他 EC2 实例那样连接到该 EC2 实例。有关更多信息，请参阅《Amazon EC2 用户指南》中的[使用 SSH 连接到您的 Linux 实例](#)。

如果指定了私有子网，您可以设置并使用堡垒主机以作为到 Hyperledger Fabric 资源的连接代理。有关更多信息，请参阅下面的[使用堡垒主机的代理连接](#)。

### Note

您可能会注意到，模板为托管 Hyperledger Fabric 服务的 EC2 实例分配了公有 IP 地址；但是，该 IP 地址不可公开访问，因为您指定的私有子网中的路由策略不允许此 IP 地址和公共源之间的流量。

## 使用堡垒主机的代理连接

包含某些配置的 Hyperledger Fabric 服务可能不会公开。在这种情况下，你可以通过堡垒主机连接到 Hyperledger Fabric 资源。有关堡垒主机的更多信息，请参阅[Linux 堡垒主机快速入门指南中的 Linux 堡垒主机架构](#)。

堡垒主机是 EC2 实例。确保满足以下要求：

- 堡垒主机的 EC2 实例位于启用了自动分配公有 IP 且具有互联网网关的公有子网内。
- 堡垒主机拥有允许 ssh 连接的密钥对。
- 堡垒主机与允许来自连接的客户端的入站 SSH 流量的安全组关联。
- 分配给 Hyperledger Fabric 主机的安全组（例如，如果 ECS 是容器平台，则分配给应用程序负载均衡器；如果 docker-local 是容器平台，则分配给主机 EC2 实例）允许 VPC 内的所有端口上的入站流量。

设置堡垒主机后，请确保连接的客户端使用堡垒主机作为代理。以下示例演示了如何使用 Mac OS 设置代理连接。*BastionIP*替换为堡垒主机 EC2 实例的 IP 地址以及*MySshKey.pem*您复制到堡垒主机的 key pair 文件。

在命令行键入以下内容：

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

这将为本地计算机上的端口 9001 设置至堡垒主机的端口转发。

接下来，将您的浏览器或系统设置为使用适用于localhost:9001 的 SOCKS 代理。例如，使用 MacOS，选择 System Preferences (系统首选项)、Network (网络)、Advanced (高级)，再选择 SOCKS proxy (SOCKS 代理)，然后键入 localhost:9001。

在 Chrome 中使用 FoxyProxy 标准版，选择“更多工具”、“扩展程序”。在“FoxyProxy 标准”下，选择“详细信息”、“扩展选项”、“添加新代理”。选择 Manual Proxy Configuration (手动代理配置)。对于 Host or IP Address (主机或 IP 地址)，键入 localhost，对于 Port (端口)，键入 9001。选择 SOCKS Proxy? (SOCKS 代理?)、Save (保存)。

您现在应能连接到模板输出中列出的 Hyperledger Fabric 主机地址。

## 文档历史记录

下表介绍了对此指南文档所做的更改。

文档最近更新时间：2019 年 5 月 1 日

更改	描述	日期
停用 Amazon Web Services Blockchain Templates。	Amazon Web Services Blockchain Templates 已于 2019 年 4 月 30 日停产。不会对本服务或本支持文档进行进一步更新。为了获得最佳的托管区块链体验 AWS，我们建议您使用 <a href="#">亚马逊托管区块链 (AMB)</a> 。	2019 年 5 月 1 日
堡垒主机更新。	修改了入门教程和以太坊添加堡垒主机的先决条件，堡垒主机允许在使用 ECS 平台时访问通过内部负载均衡器提供的 Web 资源，使用 docker-local 时允许访问 EC2 实例。	2018 年 5 月 3 日
创建指南。	支持 AWS Blockchain Templates 初始版本的全新开发人员指南。	2018 年 4 月 19 日

# AWS 词汇表

有关最新 AWS 术语，请参阅《AWS 词汇表 参考资料》中的[AWS 词汇表](#)。