aws

用户指南

Amazon Virtual Private Cloud

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon Virtual Private Cloud: 用户指南

Copyright © 2025 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

Amazon VPC 是什么？ 1
特征 1
Amazon VPC 入门 2
使用 Amazon VPC 2
Amazon VPC 的定价 3
Amazon VPC 的工作原理 5
VPC 和子网 6
默认和非默认 VPC 6
路由表 6
访问 Internet 7
访问企业或家庭网络 8
连接 VPC 和网络 8
AWS 私有全球网络 8
规划 VPC 10
注册 AWS 账户 10
验证权限 10
确定 IP 地址范围 11
选择可用区 11
规划您的互联网连接 11
创建 VPC 12
部署您的应用程序 12
IP 寻址 13
私有 IPv4 地址 14
公有 IPv4 地址 14
IPv6 地址 15
公有 IPv6 地址 16
私有 IPv6 地址 16
使用自带 IP 地址 18
使用 Amazon VPC IP 地址管理器 18
VPC CIDR 块 18
IPv4 VPC CIDR 块 19
管理 VPC 的 IPv4 CIDR 块 20
IPv4 CIDR 块关联限制 22
IPv6 VPC CIDR 块 23

子网 CIDR 块 24
IPv4 的子网定型 25
IPv6 的子网定型 25
比较 IPv4 与 IPv6 26
托管前缀列表 27
前缀列表概念和规则 28
适用于前缀列表的 Identity and Access Management 28
客户管理的前缀列表 29
AWS托管前缀列表 38
使用前缀列表优化 AWS 基础设施管理 40
AWS IP 地址范围 42
下载 43
出口控制 43
地理位置源 43
查找地址范围 44
语法 50
订阅 通知 55
VPC 的 IPv6 支持 56
为 VPC 添加 IPv6 支持 57
双堆栈 VPC 示例 61
AWS 上的 IPv6 支持 63
支持 IPv6 的服务 63
其他 IPv6 支持 71
了解更多 72
Virtual Private Cloud 73
VPC 基础知识 74
VPC IP 地址范围 74
VPC 图 74
VPC 资源 75
VPC 配置选项 76
默认 VPC 77
默认 VPC 组件 78
默认子网 80
查看默认 VPC 和默认子网 80
创建 VPC 84
创建 VPC 以及其他 VPC 资源 84

仅创建 VPC ... 86
使用 AWS CLI 创建 VPC ... 87
可视化 VPC 中的资源 ... 91
添加或删除 CIDR 块 ... 93
DHCP 选项集 ... 95
什么是 DHCP？ ... 95
DHCP 选项集概念 ... 96
使用 DHCP 选项集 ... 99
DNS 属性 ... 102
了解 Amazon DNS ... 103
查看您的 EC2 实例的 DNS 主机名称 ... 107
查看和更新 VPC 的 DNS 属性 ... 108
网络地址用量 ... 109
NAU 的计算方式 ... 109
NAU 示例 ... 110
共享 VPC 子网 ... 111
共享子网的先决条件 ... 112
使用共享子网 ... 112
拥有者与参与者的计费和计量 ... 115
所有者和参与者的责任和权限 ... 115
AWS 资源和共享 VPC 子网 ... 118
将 VPC 扩展到其他可用区 ... 119
AWS Local Zones 中的子网 ... 120
AWS Wavelength 中的子网 ... 124
AWS Outposts 中的子网 ... 127
删除您的 VPC ... 128
使用控制台删除 ... 129
使用 CLI 删除 ... 130
通过控制台操作生成 IaC ... 131
子网 ... 132
子网基础知识 ... 132
子网 IP 地址范围 ... 132
子网类型 ... 133
子网图 ... 133
子网路由 ... 134
子网设置 ... 134

子网安全性 135
创建子网 135
将 IPv6 CIDR 块添加到子网或从中删除 137
修改子网的 IP 寻址属性 138
子网 CIDR 预留 138
通过控制台使用子网 CIDR 预留 139
通过 AWS CLI 使用子网 CIDR 预留 140
路由表 141
路由表概念 142
子网路由表 143
网关路由表 149
路由优先级 151
示例路由选项 153
创建路由表和路由 167
管理子网路由表 169
替换主路由表 172
将路由表与网关关联 173
替换或还原本地路由的目标 174
VPC 中的动态路由 175
排查可达性问题 203
中间盒路由向导 203
中间盒路由向导先决条件 203
将 VPC 流量重定向到安全设备 204
中间盒路由向导注意事项 206
中间盒场景 206
删除子网 215
连接 VPC 216
互联网网关 217
互联网网关基本信息 218
创建互联网网关 220
删除互联网网关 222
仅出口互联网网关 223
仅出口互联网网关基础知识 224
向子网添加仅出口互联网访问 225
NAT 设备 228
NAT 网关 229

NAT 实例 268
比较 NAT 设备 278
弹性 IP 地址 280
弹性 IP 地址概念和规则 281
开始使用弹性 IP 地址 282
AWS Transit Gateway 290
AWS Virtual Private Network 291
VPC 对等连接 292
监控 294
VPC 流日志 295
流日志基础知识 296
流日志记录 299
流日志记录示例 308
流日志限制 315
定价 317
使用流日志 318
发布到 CloudWatch Logs 321
发布到 Amazon S3 328
发布到 Amazon Data Firehose 336
使用 Athena 查询 343
问题排查 347
CloudWatch 指标 351
NAU 指标与维度 351
启用或禁用 NAU 监控 354
NAU CloudWatch 告警示例 355
账单和使用情况报告 355
IP 地址管理 356
VPC 端点 357
中转网关 357
网络分析 358
流量镜像 358
VPC Lattice 359
跨账户/跨区域资源 359
安全性 361
数据保护 361
互联网络流量隐私 362

Identity and Access Management 363
受众 363
使用身份进行身份验证 364
使用策略管理访问 366
Amazon VPC 如何与 IAM 配合使用 368
策略示例 372
问题排查 384
AWS 托管策略 385
基础设施安全性 388
网络隔离 388
控制网络流量 388
比较安全组和网络 ACL 389
安全组 390
安全组基本信息 391
安全组示例 393
安全组规则 393
默认安全组 398
创建安全组 399
配置安全组规则 401
删除安全组 402
将安全组与多个 VPC 关联 403
与 AWS Organizations 共享安全组 406
网络 ACL 411
网络 ACL 基础知识 412
网络 ACL 规则 414
默认网络 ACL 415
自定义网络 ACL 416
路径 MTU 发现 420
创建网络 ACL 421
管理网络 ACL 关联 424
删除网络 ACL 426
示例：控制对子网中的实例的访问 427
恢复能力 430
合规性验证 431
屏蔽 VPC 和子网的公共访问权限 432
VPC BPC BPA 基础知识 432

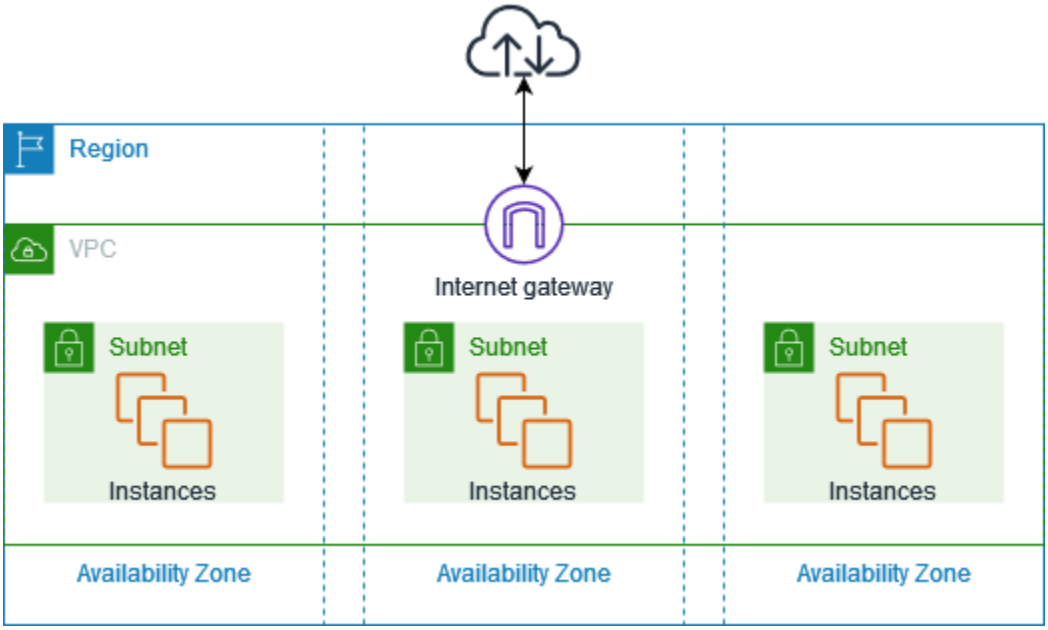
评测 VPC BPA 的影响并监控 VPC BPA 438
高级示例 442
最佳实践 494
与其他服务结合使用 495
AWS PrivateLink 495
AWS Network Firewall 497
Route 53 Resolver DNS Firewall 498
Reachability Analyzer 499
示例 501
测试环境 502
概述 502
1. 创建 VPC 504
2. 部署您的应用程序 505
3. 测试配置 505
4. 清理 505
Web 和数据库服务器 506
概述 506
1. 创建 VPC 509
2. 部署您的应用程序 510
3. 测试配置 511
4. 清理 511
私有服务器 511
概述 511
1. 创建 VPC 514
2. 部署您的应用程序 514
3. 测试配置 515
4. 清理 515
教程 516
开始通过 AWS CLI 使用 Amazon VPC 516
..... 516
先决条件 516
创建 VPC 517
创建子网 518
配置互联网连接 519
创建 NAT 网关 521
配置子网设置 522

创建安全组 522
验证您的 VPC 配置 523
部署 EC2 实例 524
故障排除 526
清理资源 527
投入生产 529
后续步骤 529
使用 AWS CLI 创建具有私有子网和 NAT 网关的 VPC 529
先决条件 516
创建 VPC 和子网 531
创建和配置互联网连接 533
创建 NAT 网关 534
为 Amazon S3 创建 VPC 端点 536
配置安全组 536
为 EC2 实例创建启动模板 537
创建负载均衡器和目标组 539
创建 自动扩缩组 540
测试配置 540
清理资源 527
后续步骤 529
配额 544
VPC 和子网 544
DNS 545
弹性 IP 地址 545
网关 545
客户管理的前缀列表 546
网络 ACL 547
网络接口 547
路由表 547
路由服务器 548
安全组 549
VPC 子网共享 550
网络地址用量 551
Amazon EC2 API 节流 551
其他配额资源 551
文档历史记录 552

Amazon VPC 是什么？

借助 Amazon Virtual Private Cloud（Amazon VPC），您可以在自己定义的逻辑隔离的虚拟网络中启动 AWS 资源。这个虚拟网络与您在数据中心中运行的传统网络极其相似，并会为您提供使用 AWS 的可扩展基础设施的优势。

下图显示了一个示例 VPC。此 VPC 在此区域的每个可用区内均有一个子网，在每个子网中有若干 EC2 实例，此外还有一个互联网网关，以允许 VPC 中资源与互联网之间进行通信。



有关更多信息，请参阅 Amazon Virtual Private Cloud (Amazon VPC)。

特征

以下功能可以帮助您配置 VPC，以提供应用程序所需的连接：

Virtual Private Cloud (VPC)

VPC 是一个虚拟网络，与您在自己的数据中心中运行的传统网络极为相似。创建 VPC 后，您可以添加子网。

子网

子网是您的 VPC 内的 IP 地址范围。子网必须位于单个可用区中。在添加子网后，您可以在 VPC 中部署 AWS 资源。

IP 寻址

您可以为您的 VPC 和子网分配 IP 地址，包括 IPv4 地址和 IPv6 地址。您还可以将您的公有 IPv4 地址和 IPv6 GUA 地址带到 AWS 并将其分配到 VPC 中的资源，例如 EC2 实例、NAT 网关和网络负载均衡器。

路由

使用路由表决定将来自您的子网或网关的网络流量定向到何处。

网关和端点

网关将您的 VPC 连到其他网络。例如，使用互联网网关将您的 VPC 连接到网络。使用 VPC 端点私下连接到 AWS 服务，无需使用互联网网关或 NAT 设备。

对等连接

使用 VPC 对等连接在两个 VPC 中的资源之间路由流量。

流量镜像

从网络接口复制网络流量，然后将其发送到安全和监控设备进行深度数据包检查。

中转网关

将中转网关用作中央枢纽，以在 VPC、VPN 连接和 AWS Direct Connect 连接之间路由流量。

Amazon VPC 流日志

流日志捕获有关在 VPC 中传入和传出网络接口的 IP 流量的信息。

VPN 连接

使用 AWS Virtual Private Network (AWS VPN) 将 VPC 连接到您的本地网络。

Amazon VPC 入门

您的 AWS 账户 在每个 AWS 区域 中包含一个默认 VPC。您的默认 VPC 已配置为可立即启动并连到 EC2 实例。有关更多信息，请参阅 规划 VPC。

您可以选择使用所需的子网、IP 地址、网关和路由创建其他 VPC。有关更多信息，请参阅 the section called “创建 VPC”。

使用 Amazon VPC

您可以使用以下任意接口创建和管理 VPC：

- AWS Management Console — 提供您可用来访问 VPC 的 Web 界面。
- AWS Command Line Interface (AWS CLI) — 提供适用于大量 AWS 服务（包括 Amazon VPC）的命令，并在 Windows、Mac 和 Linux 上得到支持。有关更多信息，请参阅 AWS Command Line Interface。
- AWS 开发工具包 — 提供特定语言的 API，并关注许多连接详细信息，例如计算签名、处理请求重试和错误处理。有关更多信息，请参阅 AWS 开发工具包。
- 查询 API — 提供您使用 HTTPS 请求调用的低级别 API 操作。使用查询 API 是用于访问 Amazon VPC 的最直接方式，但需要您的应用程序处理低级别详细信息，例如生成哈希值以签署请求以及进行错误处理。有关更多信息，请参阅《Amazon EC2 API 参考》中的 Amazon VPC 操作。

Amazon VPC 的定价

使用 VPC 无需额外付费。但是，有些 VPC 组件（如 NAT 网关、IP 地址管理器、流量镜像、Reachability Analyzer 和网络访问分析器）需要付费。有关更多信息，请参阅 Amazon VPC 定价。

您在虚拟私有云（VPC）中启动的几乎所有资源都为您提供了用于连接的 IP 地址。VPC 中的绝大多数资源都使用私有 IPv4 地址。然而，需要通过 IPv4 直接访问互联网的资源使用公有 IPv4 地址。

Amazon VPC 使您无需事先设置 VPC 即可启动弹性负载均衡、Amazon RDS 和 Amazon EMR 等托管服务。如果您有默认 VPC，它将通过在您的账户中使用它来实现此目的。托管服务为您的账户预置的任何公有 IPv4 地址都将收取费用。这些费用将与您的 AWS 成本和使用情况报告 中的 Amazon VPC 服务相关联。

公有 IPv4 地址的定价

公有 IPv4 地址是指可从互联网路由的 IPv4 地址。公有 IPv4 地址是通过 IPv4 从互联网直接访问资源所必需的。

如果您是 AWS 免费套餐的现有客户或新客户，则可以免费使用公有 IPv4 地址与 EC2 服务 750 个小时。如果您未使用 AWS 免费套餐中的 EC2 服务，则公有 IPv4 地址需要付费。有关具体定价信息，请参阅 Amazon VPC 定价中的公有 IPv4 地址选项卡。

私有 IPv4 地址 (RFC 1918) 不收费。有关共享 VPC 公有 IPv4 地址收费方式的更多信息，请参阅拥有者和参与者的计费和计量。

公有 IPv4 地址具有以下类型：

- 弹性 IP 地址 (EIP)：Amazon 提供的静态公有 IPv4 地址，您可以将其与 EC2 实例、弹性网络接口或 AWS 资源相关联。
- EC2 公有 IPv4 地址：Amazon 分配给 EC2 实例的公有 IPv4 地址（如果 EC2 实例在默认子网中启动，或者该实例在已配置为自动分配公有 IPv4 地址的子网中启动）。
- BYOIPv4 地址：您使用自带 IP 地址 (BYOIP)带到 AWS 的 IPv4 地址范围内的公有 IPv4 地址。
- 服务管理的 IPv4 地址：在 AWS 资源上自动预配置并由 AWS 服务管理的公有 IPv4 地址。例如，Amazon ECS、Amazon RDS 或 Amazon WorkSpaces 上的公有 IPv4 地址。

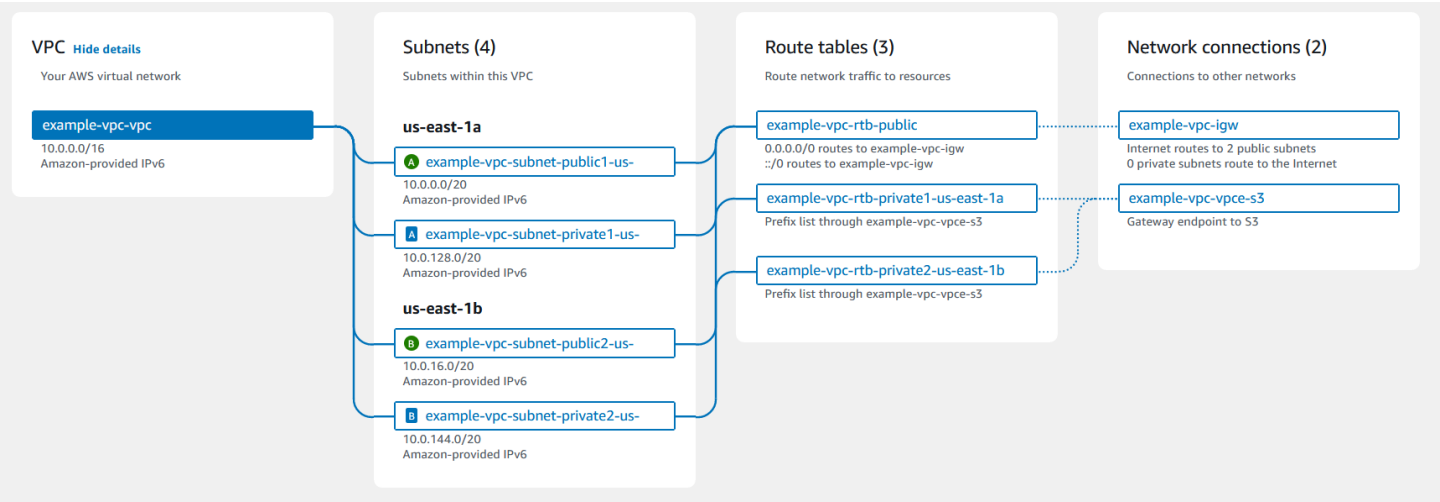
以下列表显示可以使用公有 IPv4 地址的最常见 AWS 服务列表。

- Amazon AppStream 2.0
- AWS Client VPN
- AWS Database Migration Service
- Amazon EC2
- Amazon Elastic Container Service
- Amazon EKS
- Amazon EMR
- Amazon GameLift 服务器
- AWS Global Accelerator
- AWS Mainframe Modernization
- Amazon Managed Streaming for Apache Kafka
- Amazon MQ
- Amazon RDS
- Amazon Redshift
- AWS Site-to-Site VPN
- Amazon VPC NAT 网关
- Amazon WorkSpaces
- Elastic Load Balancing

Amazon VPC 的工作原理

借助 Amazon Virtual Private Cloud（Amazon VPC），您可以在自己定义的逻辑隔离的虚拟网络中启动 AWS 资源。这个虚拟网络与您在数据中心中运行的传统网络极其相似，并会为您提供使用 AWS 的可扩展基础设施的优势。

下图直观地演示了当您使用 AWS Management Console 创建 VPC 时在预览窗格中显示的 VPC 及其资源。对于现有 VPC，您可以在资源地图选项卡上访问此可视化功能。此示例显示当您选择创建 VPC 和其他网络资源时，最初在创建 VPC 页面上选择的资源。此 VPC 在两个可用区域中配置了一个 IPv4 CIDR 和一个 Amazon 提供的 IPv6 CIDR、若干子网、三个路由表、一个互联网网关和一个网关端点。由于我们选择了互联网网关，因此图形会指示来自公有子网的流量将路由到互联网，因为相应的路由表会将流量发送到互联网网关。



概念

- VPC 和子网
- 默认和非默认 VPC
- 路由表
- 访问 Internet
- 访问企业或家庭网络
- 连接 VPC 和网络
- AWS 私有全球网络

VPC 和子网

虚拟私有云 (VPC) 是仅适用于您的 AWS 账户的虚拟网络。它在逻辑上与 AWS 云中的其他虚拟网络隔绝。您可以为 VPC 指定 IP 地址范围、添加子网、添加网关以及关联安全组。

子网是您的 VPC 内的 IP 地址范围。您可将 AWS 资源（如 Amazon EC2 实例）启动到您的子网中。您可以将子网连接到互联网、其他 VPC 和自己的数据中心，并使用路由表路由传入和传出子网的流量。

了解更多

- IP 寻址
- Virtual Private Cloud
- 子网

默认和非默认 VPC

如果您的账户是在 2013 年 12 月 4 日之后创建的，则每个区域中都有一个默认 VPC。默认 VPC 已配置且可供您使用。例如，它在区域的每个可用区中具有默认子网、已连接的互联网网关、主路由表中的路由（用于将所有流量发送到互联网网关）以及 DNS 设置（自动将公共 DNS 主机名分配给具有公有 IP 地址的实例，并通过 Amazon 提供的 DNS 服务器启用 DNS 解析）（请参阅 VPC 中的 DNS 属性）。因此，在默认子网中启动的 EC2 实例自动拥有互联网访问权限。如果您在某个区域有一个默认 VPC，并且在该区域中启动 EC2 实例时未指定子网，我们会选择一个默认子网，然后在该子网中启动实例。

您还可以创建自己的 VPC，并根据需要对其进行配置。这称为非默认 VPC。您在非默认 VPC 中创建的子网和您在默认 VPC 中创建的额外子网称为非默认子网。

了解更多信息

- the section called “默认 VPC”
- the section called “创建 VPC”

路由表

路由表包含一组称为“路由”的规则，它们用于确定将网络流量从您的 VPC 发送到何处。您可以将子网与特定路由表显式关联。否则，子网将与主路由表隐式关联。

路由表中的每个路由都指定了您希望将流量传输到的 IP 地址范围（目的地）以及发送流量所通过的网关、网络接口或连接（目标）。

了解更多信息

- 配置路由表

访问 Internet

您可以控制在 VPC 之外的 VPC 访问资源中启动实例的方式。

原定设置 VPC 包含一个互联网网关，而且每个原定设置子网都是公有子网。您在默认子网中启动的每个实例都有一个私有 IPv4 地址和一个公有 IPv4 地址。这些实例可以通过 Internet 网关与 Internet 通信。通过互联网网关，您的实例可通过 Amazon EC2 网络边界连接到 Internet。

默认情况下，您启动到非默认子网中的每个实例都有一个私有 IPv4 地址，但没有公有 IPv4 地址，除非您在启动时特意指定一个，或者修改子网的公有 IP 地址属性。这些实例可以相互通信，但无法访问 Internet。

您可以通过以下方式为在非默认子网中启动的实例启用 Internet 访问：将一个互联网网关附加到该实例的 VPC（如果其 VPC 不是默认 VPC），然后将一个弹性 IP 地址与该实例相关联。

或者，您还可以使用网络地址转换 (NAT) 设备，以允许 VPC 中的实例发起到互联网的出站连接，但阻止来自互联网的未经请求的入站连接。NAT 将多个私有 IPv4 地址映射到一个公有 IPv4 地址。您可以使用弹性 IP 地址配置 NAT 设备，并通过互联网网关将其与互联网相连。您可以通过 NAT 设备将私有子网中的实例连接到互联网，NAT 设备会将来自实例的流量路由到互联网网关，并将所有响应路由到该实例。

如果您将 IPv6 CIDR 块与 VPC 关联并为实例分配 IPv6 地址，则实例可以通过互联网网关通过 IPv6 连接到互联网。或者，实例也可以使用仅出口互联网网关经由 IPv6 发起到互联网的出站连接。IPv6 流量独立于 IPv4 流量；您的路由表必须包含单独的 IPv6 流量路由。

了解更多信息

- 使用互联网网关为 VPC 启用互联网访问
- 使用仅出口互联网网关允许出站 IPv6 流量
- 使用 NAT 设备连接到互联网或其他网络

访问企业或家庭网络

您可以选择使用 IPsec AWS Site-to-Site VPN 连接将您的 VPC 与公司的数据中心连接，从而将 AWS Cloud 作为数据中心的延伸。

Site-to-Site VPN 连接由 AWS 端的虚拟私有网关或中转网关与位于数据中心的客户网关设备之间的两条 VPN 隧道组成。客户网关设备是站点到站点 VPN 连接在您这一端配置的实体设备或软件设备。

了解更多

- 《AWS Site-to-Site VPN 用户指南》
- Amazon VPC 中转网关

连接 VPC 和网络

您可以在两个 VPC 之间创建一个 VPC 对等连接，然后通过此连接不公开地在这两个 VPC 之间路由流量。这两个 VPC 中的实例可以彼此通信，就像它们在同一网络中一样。

您还可以创建一个中转网关，并使用它来互连 VPC 和本地网络。中转网关充当区域虚拟路由器，用于其各种连接（可包括 VPC、VPN 连接、AWS Direct Connect 网关和中转网关对等连接）之间的流量流动。

了解更多

- Amazon VPC Peering Guide
- Amazon VPC 中转网关

AWS 私有全球网络

AWS 以其高性能、低延迟的私有全球网络，提供安全的云计算环境以支持您的网络需求。AWS区域连接到多个互联网服务提供商 (ISP) 以及私有全球网络主干，从而为客户发送的跨区域流量提供改进的网络性能。

源自私有全球网络且目标位于私有全球网络中的数据包将保留在私有全球网络中，不会穿越公共互联网。无论目标是私有 IP 地址还是公有 IP 地址，都是如此。例如，如果两个 VPC 中的 EC2 实例使用公有 IP 地址进行通信，则流量将保留在私有全球网络中。目标可以位于同一个可用区、同一区域中的不同可用区，也可以位于不同区域（中国区域除外）。

网络数据包丢失可能因多种因素导致，包括网络流碰撞、低级（第 2 层）错误和其他网络故障。我们设计并运行我们的网络以最大限度地减少数据包丢失。我们跨连接 AWS 区域的全球骨干网衡量数据包丢失率 (PLR)。我们运营我们的骨干网络，目标是使 p99 达到每小时 PLR 低于 0.0001%。

规划 VPC

完成以下任务以准备好创建和连接 VPC。完成后，即可在 AWS 上部署应用程序。

任务

- 注册 AWS 账户
- 验证权限
- 确定 IP 地址范围
- 选择可用区
- 规划您的互联网连接
- 创建 VPC
- 部署您的应用程序

注册 AWS 账户

如果您还没有 AWS 账户，请完成以下步骤来创建一个。

注册 AWS 账户

1. 打开 https://portal.aws.amazon.com/billing/signup。
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 AWS 账户 时，系统将会创建一个 AWS 账户根用户。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。

注册过程完成后，AWS 会向您发送一封确认电子邮件。在任何时候，您都可以通过转至 https://aws.amazon.com/ 并选择我的账户来查看当前的账户活动并管理您的账户。

验证权限

您必须具备所需的权限，才能使用 Amazon VPC。有关更多信息，请参阅适用于 Amazon VPC 的 Identity and Access Management和Amazon VPC 策略示例。

确定 IP 地址范围

VPC 中的资源将使用 IP 地址相互通信以及与互联网上的资源进行通信。创建 VPC 和子网时，您可以选择其 IP 地址范围。当您在子网中部署资源（例如 EC2 实例）时，这些资源将获得来自子网 IP 地址范围内的 IP 地址。有关更多信息，请参阅 IP 寻址。

在选择 VPC 的大小时，请考虑在您的 AWS 账户和 VPC 中将需要的 IP 地址数量。确保 VPC 的 IP 地址范围不会与您自己网络的 IP 地址范围重叠。如果您需要在多个 VPC 之间建立连接，则必须确保其 IP 地址没有重叠。

IP 地址管理器（IPAM）可让您更轻松地计划、跟踪和监控应用程序的 IP 地址。有关更多信息，请参阅 IP 地址管理器指南。

选择可用区

AWS 区域是我们集中管理数据中心的物理位置，称为可用区。每个可用区都具有独立的电源、制冷和物理安防设施，并且配备了冗余电源、联网和连接。一个区域中的可用区在物理上隔离并且相互保持较远的距离，相互通过高带宽、低延迟的网络连接。您可以将应用程序设计为在多个可用区中运行，以实现更高的容错能力。

生产环境

对于生产环境，我们建议您至少选择两个可用区，并在每个活动可用区内均衡部署您的 AWS 资源。

开发或测试环境

对于开发或测试环境，您可以选择仅在一个可用区中部署资源以节省费用。

规划您的互联网连接

根据您的连接需求，规划将每个 VPC 划分为若干子网。例如：

- 如果您有 Web 服务器将接收来自互联网的客户端流量，请在每个可用区为这些服务器创建一个子网。
- 如果您还有服务器将仅接收来自 VPC 中其他服务器的流量，请在每个可用区为这些服务器创建一个单独的子网。
- 如果您有服务器将仅接收通过 VPN 连接发送到您的网络的流量，请在每个可用区为这些服务器创建一个单独的子网。

如果您的应用程序将接收来自互联网的流量，则 VPC 必须具有互联网网关。将互联网网关附加到 VPC 并不会自动使您的实例可从互联网访问。除了附加互联网网关外，您必须使用指向互联网网关的路由更新子网路由表。您还必须确保实例具有一个公有 IP 地址和一个关联的安全组，该安全组允许来自互联网的流量通过您的应用程序所要求的特定端口和协议。

您也可以将实例注册到面向互联网的负载均衡器。负载均衡器将接收来自客户端的流量，并在一个或多个可用区内的注册实例之间分配流量。有关更多信息，请参阅 Elastic Load Balancing。要允许私有子网中的实例访问互联网（例如，为了下载更新），但不允许来自互联网的未经请求的入站连接，请在每个活动可用区中添加一个公有 NAT 网关，并更新路由表以将互联网流量发送到该 NAT 网关。有关更多信息，请参阅 the section called “从私有子网访问互联网”。

创建 VPC

确定好所需的 VPC 和子网数量、要分配给 VPC 和子网的 CIDR 块以及将 VPC 连接到互联网的方式后，即可以创建您的 VPC。如果您使用 AWS Management Console 创建 VPC 并在配置中包含公有子网，我们会为该子网创建路由表并添加直接访问互联网所需的路由。有关更多信息，请参阅 the section called “创建 VPC”。

部署您的应用程序

创建 VPC 后，您可以部署应用程序。

生产环境

对于生产环境，您可以使用下列服务中的一种在多个可用区部署服务器、配置扩缩以保持应用程序所需的最低服务器数量，并将服务器注册到负载均衡器以在服务器之间均匀分配流量。

- Amazon EC2 Auto Scaling
- EC2 Fleet
- Amazon Elastic Container Service（Amazon ECS）

开发或测试环境

对于开发或测试环境，您可以选择启动单个 EC2 实例。有关更多信息，请参阅《Amazon EC2 用户指南》中的 Amazon EC2 入门。

为 VPC 和子网分配 IP 地址

IP 地址使 VPC 中的资源能够相互通信以及与 Internet 上的资源进行通信。

无类别域间路由（CIDR）表示法是一种表示 IP 地址及其网络掩码的方法。这些地址的格式如下：

- 单个 IPv4 地址为 32 位，分为 4 组，每组包含最多 3 个十进制数字（0-255）。例如：10.0.1.0。
- IPv4 CIDR 块包含一个 IPv4 地址，后跟斜杠和一个介于 0 到 32 的数字。例如，10.0.0.0/16 表示介于 10.0.0.0 到 10.0.255.255 之间的 65536 个 IPv4 地址。
- 单个 IPv6 地址为 128 位，分为 8 段，每段包含最多 4 个十六进制数字。例如：2001:0db8:85a3:0000:0000:8a2e:0370:7334。分段无需包含前导零。您也可以在地址中用双冒号（::）替换连续的全零分段。因此，示例地址可以压缩为 2001:db8:85a3::8a2e:370:7334。
- IPv6 CIDR 块中的 IPv6 地址以全零分段结尾（全零分段已替换为双冒号），后跟斜杠和一个介于 0 到 128 之间的数字。例如，2001:db8:1234:1a00::/56 表示介于 2001:db8:1234:1a00:0000:0000:0000:0000 到 2001:db8:1234:1aff:ffff:ffff:ffff:ffff 之间的 2^72 个 IPv6 地址。

有关更多信息，请参阅什么是 CIDR？

内容

- 私有 IPv4 地址
- 公有 IPv4 地址
- IPv6 地址
- 使用自带 IP 地址
- 使用 Amazon VPC IP 地址管理器
- VPC CIDR 块
- 子网 CIDR 块
- 比较 IPv4 与 IPv6
- 使用托管前缀列表合并和管理网络 CIDR 块
- AWS IP 地址范围
- VPC 的 IPv6 支持
- 支持 IPv6 的 AWS 服务

私有 IPv4 地址

私有 IPv4 地址（在本主题中也称作私有 IP 地址）无法通过 Internet 访问，但可用于 VPC 中实例之间的通信。当您在 VPC 中启动实例时，系统会将子网 IPv4 地址范围内的主要私有 IP 地址分配给实例的主要网络接口（例如，eth0）。另外，还为每个实例指定一个可解析为实例私有 IP 地址的私有（内部）DNS 主机名。主机名可以有两种类型：基于资源或基于 IP。有关更多信息，请参阅 EC2 实例命名。如果您未指定主要私有 IP 地址，我们会在子网范围内为您选择可用的 IP 地址。有关网络接口的更多信息，请参阅《Amazon EC2 用户指南》中的弹性网络接口。

您可以为 VPC 中运行的实例分配其他私有 IP 地址，即所谓的辅助私有 IP 地址。与主要私有 IP 地址不同的是，您可以将一个网络接口的辅助私有 IP 地址重新分配给另一个网络接口。私有 IP 地址会在实例停止并重新启动时保持与网络接口的关联，并在实例终止时释放。有关主要和辅助 IP 地址的更多信息，请参阅《Amazon EC2 用户指南》的多个 IP 地址。

我们所说的私有 IP 地址是 VPC 的 IPv4 CIDR 范围内的 IP 地址。大部分 VPC IP 地址范围均处于 RFC 1918 中指定的私有（非公有可路由）IP 地址范围内；但是，您可为您的 VPC 使用公有可路由的 CIDR 块。不管您的 VPC 使用何种 IP 地址范围，我们都不支持从您的 VPC 的 CIDR 块（包括公共可路由的 CIDR 块）直接访问 Internet。您必须通过网关设置 Internet 访问，例如，通过互联网网关、虚拟专用网关、AWS Site-to-Site VPN 连接或 AWS Direct Connect。

我们永远不会向互联网传播子网的 IPv4 地址范围。

公有 IPv4 地址

所有子网都有一个用于确定在子网中创建的网络接口是否自动接收公有 IPv4 地址（在本主题中也称作公有 IP 地址）的属性。因此，当您在启用了此属性的子网中启动实例时，系统会向为此实例创建的主网络接口分配一个公有 IP 地址。公有 IP 地址通过网络地址转换 (NATI) 映射到主要私有 IP 地址。

> **Note**
>
> AWS 将对所有公有 IPv4 地址收费，包括与运行的实例相关联的公有 IPv4 地址和弹性 IP 地址。有关更多信息，请参阅 Amazon VPC 定价页面中的公有 IPv4 地址定价选项卡。

您可以通过执行以下操作，控制实例是否接收公有 IP 地址：

- 修改子网的公有 IP 寻址属性。有关更多信息，请参阅 修改子网的 IP 寻址属性。
- 在实例启动过程中启用或禁用公有 IP 寻址功能，以覆盖子网的公有 IP 寻址属性。

- 启动后，您可以通过管理与网络接口关联的 IP 地址来将实例的公有 IP 地址取消分配。有关更多信息，请参阅《Amazon EC2 用户指南》中的管理 IP 地址。

公有 IP 地址将从 Amazon 的公有 IP 地址池分配，它不与您的账户关联。在公有 IP 地址与您的实例取消关联后，该地址即释放回该池，并且不再可供您使用。在某些情况下，我们会从您的实例释放公有 IP 地址，或为其分配新地址。有关更多信息，请参阅《Amazon EC2 用户指南》中的公有 IP 地址。

如果您需要向您的账户分配一个永久公有 IP 地址（您可根据需要将其分配给实例或将其从实例中删除），请改为使用弹性 IP 地址。有关更多信息，请参阅 将弹性 IP 地址关联到 VPC 中的资源。

如果您的 VPC 启用了对 DNS 主机名的支持，则系统还会向收到公有 IP 地址或弹性 IP 地址的每个实例分配一个公有 DNS 主机名。我们会将公有 DNS 主机名解析为该实例在实例网络外的公有 IP 地址和在实例网络内的私有 IP 地址。有关更多信息，请参阅 VPC 中的 DNS 属性。

如果使用的是 Amazon VPC IP 地址管理器（IPAM），则可以从 AWS 中获取连续的公有 IPv4 地址块，再用其将连续弹性 IP 地址分配给 AWS 资源。使用连续的 IPv4 地址块可以显著减少安全访问控制列表的管理开销，简化在 AWS 上扩展的企业的 IP 地址分配和跟踪工作。有关更多信息，请参阅《Amazon VPC IPAM User Guide》中的 Allocate sequential Elastic IP addresses from an IPAM pool。

IPv6 地址

随着互联网的持续增长，对 IP 地址的需求也在增加。IP 地址最常见的格式是 IPv4。IP 地址的新格式是 IPv6，它可提供比 IPv4 更大的地址空间。IPv6 解决了 IPv4 地址耗尽的问题，有助将更多设备连接到互联网。这是个渐进的过渡过程，不过随着 IPv6 采用率的提高，您可以简化网络并利用 IPv6 的高级功能来改善连接性、性能和安全性。

许多 AWS 服务（例如 Amazon EC2、Amazon S3 和 Amazon CloudFront）都提供双堆栈（IPv4 和 IPv6）或仅支持 IPv6 的选项，允许资源被分配到 IPv6 地址并通过 IPv6 协议进行访问，从而为采用 IPv6 的客户简化网络配置和管理。其他服务提供有限或部分双堆栈和仅支持 IPv6。

有关支持 IPv6 的服务的更多信息，请参阅支持 IPv6 的 AWS 服务。

请注意，国际互联网工程任务组保留了部分 IPv6 地址。有关预留 IPv6 地址范围的更多信息，请参阅 IANA IPv6 特殊用途地址注册表和 RFC4291。

> **Note**
>
> AWS 同时提供公有和私有 IPv6 寻址。AWS 认为公有 IP 地址是从 AWS 公开发布在互联网上的，而私有 IP 地址不是，也不能从 AWS 公开发布在互联网上。

内容

- 公有 IPv6 地址
- 私有 IPv6 地址

公有 IPv6 地址

Amazon 提供的 IPv6 地址始终在互联网上公开公布。IPv6 地址具有全局唯一性，因此可通过互联网访问。您可以通过控制子网的路由或通过使用安全组和网络 ACL 来控制能否通过实例的 IPv6 地址访问 EC2 实例等资源。

以下是可将公有 IPv6 地址用于工作负载的一些方法：

- 使用 Amazon VPC IP 地址管理器创建 IPAM，并向 IPAM 地址池预置 Amazon 拥有的公有 IPv6 地址范围。有关更多信息，请参阅《Amazon VPC IPAM 用户指南》中的创建 IPv6 池。
- 如果已有 IPAM 并且拥有公有 IPv6 地址范围，则可将部分或全部公有 IPv6 地址范围引入 IPAM，并向 IPAM 地址池预置公有 IPv6 地址范围。有关更多信息，请参阅《Amazon VPC IPAM 用户指南》中的教程：将 IP 地址带入 IPAM。
- 如果没有 IPAM，但拥有公有 IPv6 地址范围，则可将部分或全部公有 IPv6 地址范围引入 AWS。有关更多信息，请参阅《Amazon EC2 用户指南》中的在 Amazon EC2 中使用您自己的 IP 地址（BYOIP）。

准备好使用公有 IPv6 地址后，就可以为实例分配公有 IPv6 地址（请参阅《Amazon EC2 用户指南》中的 IPv6 地址），为 VPC 分配公有 IPv6 CIDR 块（请参阅将 CIDR 块添加到 VPC 或从中删除），并将该 IPv6 CIDR 块与子网关联（请参阅修改子网的 IP 寻址属性）。

私有 IPv6 地址

私有 IPv6 地址指并未且不能从 AWS 公开发布在互联网上的 IPv6 地址。

如果希望自己的私有网络支持 IPv6，并且不打算将流量从这些地址路由到互联网，则可使用私有 IPv6 地址。如果要从具有私有 IPv6 地址的资源连接到互联网，必须通过另一个子网中具有公有 IPv6 地址的资源路由流量，才能实现该目的。

私有 IPv6 地址有两种类型：

- IPv6 ULA 范围：RFC4193 中定义的 IPv6 地址。这些地址范围总是以“fc”或“fd”开头，因此很容易识别。有效的 IPv6 ULA 空间是指任何低于 fd00:: /8 且不与 Amazon 预留范围 fd00::/16 重叠的空间。
- IPv6 GUA 范围：RFC3587 中定义的 IPv6 地址。使用 IPv6 GUA 范围作为私有 IPv6 地址的选项默认处于禁用状态，必须在启用后才能使用。有关更多信息，请参阅《Amazon VPC IPAM 用户指南》中的启用预置私有 IPv6 GUA CIDR。

请注意以下几点：

- 私有 IPv6 地址只能通过 Amazon VPC IP 地址管理器（IPAM）使用。IPAM 发现具有 IPv6 ULA 和 GUA 地址的资源，并监控池中是否存在重叠的 IPv6 ULA 和 GUA 地址空间。
- 在使用私有 IPv6 GUA 范围时，要求使用自己拥有的 IPv6 GUA 范围。
- AWS 不会也不能在互联网上公开发布私有 IPv6 地址。即使 VPC 中有互联网网关或仅限出口的互联网网关，AWS 也不允许从私有 IPv6 地址范围直接出口到公共互联网。私有 IPv6 地址会被自动丢弃在互联网网关边缘，确保不会被公开路由。
- AWS 会保留前四个以及最后一个子网私有 IPv6 地址。
- 私有 IPv6 ULA 的有效范围为 /9 至 /60（从 fd80::/9 开始）。
- 如果已为 VPC 分配私有 IPv6 GUA 范围，则不能使用与同一 VPC 中私有 IPv6 GUA 空间重叠的公有 IPv6 GUA 空间。
- 支持具有私有 IPv6 ULA 及 GUA 地址范围的资源之间的通信（例如跨 Direct Connect、VPC 对等连接、中转网关或 VPN 连接）。
- 您可以将私有 IPv6 地址用于仅限 IPv6 和双栈 VPC 子网、弹性负载均衡器和 AWS Global Accelerator 端点。
- 使用私有 IPv6 地址不会产生任何费用。

以下是可将私有 IPv6 地址用于工作负载的一些方法：

- 使用 Amazon VPC IP 地址管理器创建 IPAM，并向 IPAM 地址池预置私有 IPv6 ULA 范围。有关更多信息，请参阅《Amazon VPC IPAM 用户指南》中的创建 IPv6 池。

- 使用 Amazon VPC IP 地址管理器创建 IPAM，并为 IPAM 地址池预置私有 IPv6 GUA 范围。使用 IPv6 GUA 范围作为私有 IPv6 地址的选项默认处于禁用状态，必须在 IPAM 上启用后才能使用。有关更多信息，请参阅《Amazon VPC IPAM 用户指南》中的启用预置私有 IPv6 GUA CIDR。

准备好使用私有 IPv6 地址后，就可以为 VPC 分配 IPAM 池中的私有 IPv6 CIDR 块（请参阅将 CIDR 块添加到 VPC 或从中删除），并将该 IPv6 CIDR 块与子网关联（请参阅修改子网的 IP 寻址属性）。

使用自带 IP 地址

您可以将部分或全部自带公有 IPv4 地址或 IPv6 地址范围引入到您的 AWS 账户。您继续拥有该地址范围，但 AWS 默认将其发布到 Internet 上。在将地址范围引入 AWS 中之后，它会在您的账户中显示为地址池。您可以从 IPv4 地址池创建弹性 IP 地址，也可以将 IPv6 地址池中的 IPv6 CIDR 块与 VPC 相关联。

有关更多信息，请参阅《Amazon EC2 用户指南》中的自带 IP 地址（BYOIP）。

使用 Amazon VPC IP 地址管理器

Amazon VPC IP 地址管理器 (IPAM) 是一项 VPC 功能，可让您更轻松地计划、跟踪和监控 AWS 工作负载的 IP 地址。您可以使用特定的业务规则用 IPAM 将 IP 地址 CIDR 分配给 VPC。

有关更多信息，请参阅《Amazon VPC IPAM 用户指南》中的什么是 IPAM？。

VPC CIDR 块

您的虚拟私有云（VPC）的 IP 地址以无类别域间路由（CIDR）表示法表示。VPC 必须具有一个关联的 IPv4 CIDR 块。您可以选择性地关联其他 IPv4 CIDR 块和一个或多个 IPv6 CIDR 块。有关更多信息，请参阅 为 VPC 和子网分配 IP 地址。

内容

- IPv4 VPC CIDR 块
- 管理 VPC 的 IPv4 CIDR 块
- IPv4 CIDR 块关联限制
- IPv6 VPC CIDR 块

IPv4 VPC CIDR 块

当您创建 VPC 时，必须为这个 VPC 指定 IPv4 CIDR 块。允许的块大小介于 /16 网络掩码 (65,536 个 IP 地址) 和 /28 网络掩码 (16 个 IP 地址) 之间。在创建 VPC 后，您可以将额外的 IPv4 CIDR 块与 VPC 关联。有关更多信息，请参阅 将 CIDR 块添加到 VPC 或从中删除。

在创建 VPC 时，建议您指定来自私有 IPv4 地址范围的 CIDR 块（如 RFC 1918 中所指定）。

RFC 1918 范围	示例 CIDR 块
10.0.0.0 - 10.255.255.255（10/8 前缀）	10.0.0.0/16
172.16.0.0 - 172.31.255.255（172.16/12 前缀）	172.31.0.0/16
192.168.0.0 - 192.168.255.255（192.168/16 前缀）	192.168.0.0/20

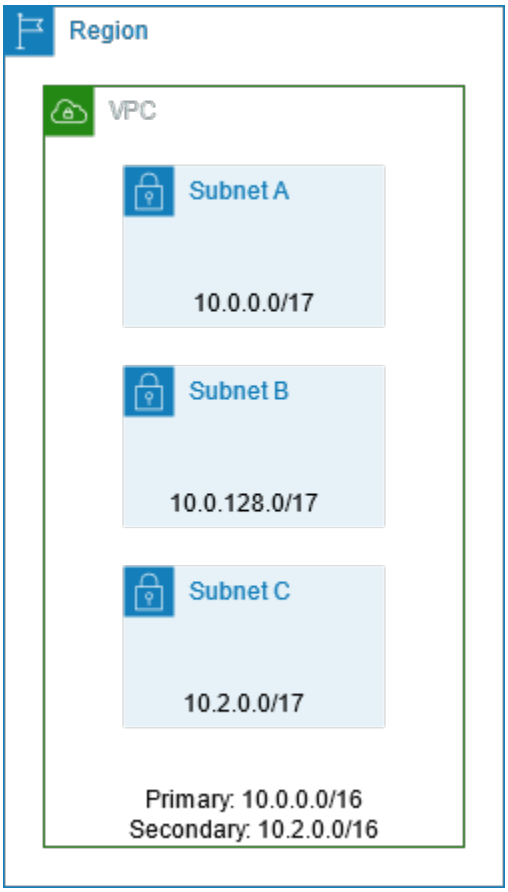
注意事项

- 您无法为您的 VPC 指定以下 CIDR 块：
  - 0.0.0.0/8
  - 127.0.0.0/8（内部主机环回地址范围）
  - 169.254.0.0/16（链路本地地址范围）
  - 224.0.0.0/4（组播地址范围）
- 创建 VPC 以用于 AWS 服务时，请参阅服务文档以验证其配置是否有特定要求。
- 某些 AWS 服务使用 `172.17.0.0/16` CIDR 范围。如果 IP 地址范围已在网络中使用，则服务可能会遇到 IP 地址冲突。例如，AWS Cloud9 和 Amazon SageMaker AI 使用 `172.17.0.0/16`。为避免发生冲突，请不要在创建 VPC 时使用此范围。有关更多信息，请参阅《AWS Cloud9 用户指南》中的无法连接到 EC2 环境，因为 VPC 的 IP 地址被 Docker 使用。
- 您可以创建一个具有公共可路由的 CIDR 块 (不在 RFC 1918 中指定的私有 IPv4 地址范围内) 的 VPC。但是，出于本文档的写作目的，我们的私有 IP 地址指的是位于 VPC 的 CIDR 范围内的 IPv4 地址。
- 如果您使用命令行工具或 Amazon EC2 API 创建 VPC，则系统会自动将 CIDR 块修改为其规范形式。例如，假设您为 CIDR 块指定 100.68.0.18/18，我们将创建一个 CIDR 块 100.68.0.0/18。

管理 VPC 的 IPv4 CIDR 块

您可以将辅助 IPv4 CIDR 块与 VPC 关联。当您将 CIDR 块与 VPC 关联时，路由会自动添加到 VPC 路由表中，以便在 VPC 中启用路由 (目的地是 CIDR 块，目标是 `local`)。

在下面的示例中，VPC 同时具有一个主 CIDR 块和一个辅助 CIDR 块。子网 A 和子网 B 的 CIDR 块来自主 VPC CIDR 块。子网 C 的 CIDR 块来自辅助 VPC CIDR 块。



下面的路由表显示了 VPC 的本地路由。

目标位置	目标
10.0.0.0/16	本地
10.2.0.0/16	本地

要将 CIDR 块添加到 VPC，应遵循以下规则：

- 允许的块大小在 /28 网络掩码与 /16 网络掩码之间。

- 该 CIDR 块不得与 VPC 所关联的任何现有 CIDR 块重叠。
- 您可以使用的 IPv4 地址范围是有限制的。有关更多信息，请参阅 IPv4 CIDR 块关联限制。
- 您不能增加或减少现有 CIDR 块的大小。
- 可以与 VPC 关联的 CIDR 块数和可以添加到路由表的路由数是有配额的。如果这导致您超出配额，您就不能关联 CIDR 块。有关更多信息，请参阅 Amazon VPC 配额。
- CIDR 块不得与任何 VPC 路由表中的路由中的 CIDR 范围相同或大于该范围。例如，在主要 CIDR 块为 `10.2.0.0/16` 的 VPC 中，您的路由表中有一个指向虚拟私有网关的现有路由，目的地为 `10.0.0.0/24`。您要关联 `10.0.0.0/16` 范围内的辅助 CIDR 块。由于该现有路由，您无法关联 `10.0.0.0/24` 或更大的 CIDR 块。但是，您可以关联 `10.0.0.0/25` 或更小的辅助 CIDR 块。
- 在向作为 VPC 对等连接的一部分的 VPC 中添加 IPv4 CIDR 块时，应遵循以下规则：
  - 如果 VPC 对等连接为 `active`，则可以向 VPC 中添加 CIDR 块，条件是这些块不与对等 VPC 的 CIDR 块重叠。
  - 如果 VPC 对等连接为 `pending-acceptance`，则请求方 VPC 的拥有者不能向 VPC 中添加任何 CIDR 块，无论它是否与接受方 VPC 的 CIDR 块重叠。要么接受方 VPC 的拥有者必须接受对等连接，要么请求方 VPC 的拥有者必须删除 VPC 对等连接请求，添加 CIDR 块，然后请求新的 VPC 对等连接。
  - 如果 VPC 对等连接为 `pending-acceptance`，则接受方 VPC 的拥有者可以向 VPC 中添加 CIDR 块。如果辅助 CIDR 块与请求方 VPC 的 CIDR 块重叠，则 VPC 对等连接请求将失败，无法被接受。
- 如果您使用 AWS Direct Connect 来通过 Direct Connect 网关连接到多个 VPC，则与 Direct Connect 网关关联的 VPC 不得具有重叠的 CIDR 块。如果您将 CIDR 块连接到其中一个与 Direct Connect 网关关联的 VPC，请确保新的 CIDR 块不会与任何其他关联 VPC 的现有 CIDR 块重叠。有关更多信息，请参阅《AWS Direct Connect 用户指南》中的 Direct Connect 网关。
- 在添加或删除 CIDR 块时，它会经历不同的状态：`associating` |`associated` |`disassociating` |`disassociated` |`failing` |`failed` 。当 CIDR 块处于 `associated` 状态时，表示它已准备就绪，可供您使用。

您可以取消与 VPC 相关联的 CIDR 块的关联，但无法取消最初用于创建 VPC 的 CIDR 块 (主要 CIDR 块) 的关联。要在 Amazon VPC 控制台中查看 VPC 的主要 CIDR，请选择 Your VPCs (您的 VPC) ，选中您的 VPC 的复选框，然后再选择 CIDR 选项卡。要使用 AWS CLI 查看主要 CIDR，请使用 describe-vpcs 命令，如下所示。主要 CIDR 返回到高级别 `CidrBlock` `element`。

```
aws ec2 describe-vpcs --vpc-id vpc-1a2b3c4d --query Vpcs[*].CidrBlock --output text
```

下面是示例输出。

```
10.0.0.0/16
```

IPv4 CIDR 块关联限制

下表概述了允许和受限的 VPC CIDR 块关联。限制的原因是某些 AWS 服务使用了跨 VPC 和跨账户功能，这些功能需要在 AWS 服务端使用无冲突的 CIDR 块。

IP 地址范围	受限制的关联	允许的关联
10.0.0.0/8	其他 RFC 1918* 范围 (172.16.0.0/12 和 192.168.0.0/16) 中的 CIDR 块。 如果与 VPC 关联的任何 CIDR 块属于 10.0.0.0/15 范围（10.0.0.0 至 10.1.255.255），则不能添加属于 10.0.0.0/16 范围（10.0.0.0 至 10.0.255.255）的 CIDR 块。 198.19.0.0/16 范围中的 CIDR 块。	10.0.0.0/8 范围中介于 /16 网络掩码和 /28 网络掩码之间不受限制的任何其他 CIDR 块。 100.64.0.0/10 范围中介于 /16 网络掩码和 /28 网络掩码之间任何可公开路由的 IPv4 CIDR 块（非 RFC 1918）或 /16 网络掩码和 /28 网络掩码之间的 CIDR 块。
169.254.0.0/16	如 RFC 5735 中所述，来自“链路本地”块的 CIDR 块属于保留块，不能分配给 VPC。	
172.16.0.0/12	其他 RFC 1918* 范围 (10.0.0.0/8 和 192.168.0.0/16) 中的 CIDR 块。 172.31.0.0/16 范围中的 CIDR 块。 198.19.0.0/16 范围中的 CIDR 块。	172.16.0.0/12 范围中介于 /16 网络掩码和 /28 网络掩码之间不受限制的任何其他 CIDR 块。 100.64.0.0/10 范围中介于 /16 网络掩码和 /28 网络掩码之间任何可公开路由的 IPv4 CIDR 块（非 RFC 1918）或 /16 网络掩码和 /28 网络掩码之间的 CIDR 块。
192.168.0.0/16	其他 RFC 1918* 范围（10.0.0.0/8 和 172.16.0.0/12）中的 CIDR 块。	192.168.0.0/16 范围中介于 /16 网络掩码和 /28 网络掩码之间的任何其他 CIDR 块。

IP 地址范围	受限制的关联	允许的关联
	198.19.0.0/16 范围中的 CIDR 块。	100.64.0.0/10 范围中介于 /16 网络掩码和 /28 网络掩码之间任何可公开路由的 IPv4 CIDR 块（非 RFC 1918）或 /16 网络掩码和 /28 网络掩码之间的 CIDR 块。
198.19.0.0/16	RFC 1918* 范围中的 CIDR 块。	100.64.0.0/10 范围中介于 /16 网络掩码和 /28 网络掩码之间任何可公开路由的 IPv4 CIDR 块（非 RFC 1918）或 /16 网络掩码和 /28 网络掩码之间的 CIDR 块。
可公开路由的 CIDR 块（非 RFC 1918），或 100.64.0.0/10 范围中的 CIDR 块。	RFC 1918* 范围中的 CIDR 块。 198.19.0.0/16 范围中的 CIDR 块。	100.64.0.0/10 范围中介于 /16 网络掩码和 /28 网络掩码之间任何其他可公开路由的 IPv4 CIDR 块（非 RFC 1918）或 /16 网络掩码和 /28 网络掩码之间的 CIDR 块。 您也可以在其中一个 RFC 1918 范围内关联 CIDR，但是要做到这一点，您必须在创建 VPC 时先添加该 CIDR，然后再添加非 RFC 1918 CIDR。

*RFC 1918 范围是指 RFC 1918 中指定的私有 IPv4 地址范围。

IPv6 VPC CIDR 块

创建新 VPC 时，您可以关联一个 IPv6 CIDR 块；您可以关联最多 5 个 IPv6 CIDR 块，从 /44 到 /60，增量为 /4。您可以从 Amazon 的 IPv6 地址池请求 IPv6 CIDR 块。有关更多信息，请参阅 将 CIDR 块添加到 VPC 或从中删除。

如果您已向 VPC 关联 IPv6 CIDR 块，则可以将 IPv6 CIDR 块与 VPC 中的现有子网关联，或在创建新子网时执行此操作。有关更多信息，请参阅 the section called “IPv6 的子网定型”。

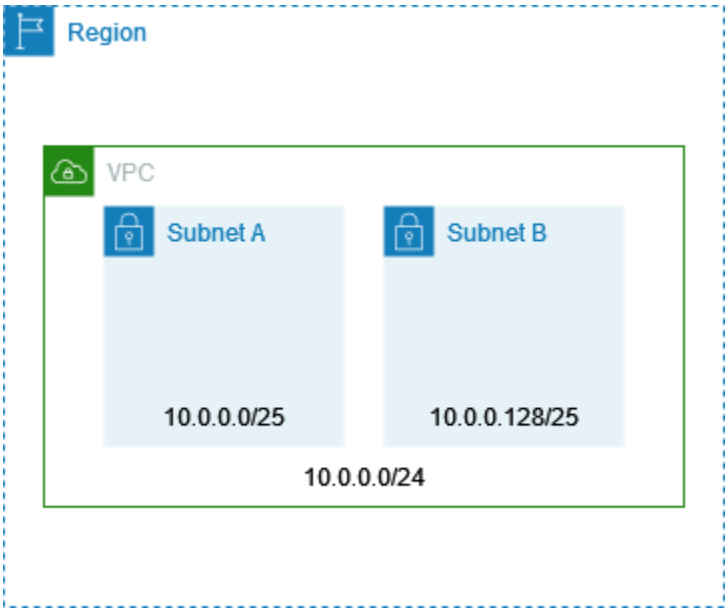
例如，您可以创建一个 VPC 并指定要向此 VPC 关联 Amazon 提供的 IPv6 CIDR 块。Amazon 向您的 VPC 分配以下 IPv6 CIDR 块：2001:db8:1234:1a00::/56。无法自行选择 IP 地址范围。您可以创建一个子网并从此范围分配 IPv6 CIDR 块；例如，2001:db8:1234:1a00::/64。

您可以取消 IPv6 CIDR 块与 VPC 的关联。在取消 IPv6 CIDR 块与 VPC 的关联后重新关联它们时，不一定会收到相同的 CIDR 块。

子网 CIDR 块

您的子网的 IP 地址以无类别域间路由（CIDR）表示法表示。子网的 CIDR 块可以与 VPC 的 CIDR 块相同（用于在 VPC 中创建单个子网），也可以是 VPC 的 CIDR 块的一个子集（用于在 VPC 中创建多个子网）。如果您在 VPC 中创建多个子网，子网的 CIDR 块不能重叠。

例如，如果创建其 CIDR 块为 10.0.0.0/24 的 VPC，则它支持 256 个 IP 地址。您可以将这个 CIDR 块分散到两个子网，每个子网支持 128 个 IP 地址。一个子网使用 CIDR 块 10.0.0.0/25 (对于地址 10.0.0.0 - 10.0.0.127)，另一个子网使用 CIDR 块 10.0.0.128/25 (对于地址 10.0.0.128 - 10.0.0.255)。



互联网上提供的工具可帮助您计算和创建 IPv4 和 IPv6 子网 CIDR 块。您可以通过搜索“子网计算器”或“CIDR 计算器”等术语来找到满足您需求的工具。您的网络工程组也可以帮助您判断可为子网指定哪些具体 IPv4 和 IPv6 CIDR 块。

IPv4 的子网定型

子网允许的 IPv4 CIDR 块大小在 /28 网络掩码与 /16 网络掩码之间。每个子网 CIDR 块中的前四个 IP 地址和最后一个 IP 地址无法供您使用，而且无法分配给任何资源（例如 EC2 实例）。例如，在具有 CIDR 块 10.0.0.0/24 的子网中，以下五个 IP 地址是保留的：

- 10.0.0.0：网络地址。
- 10.0.0.1：由 AWS 保留，用于 VPC 路由器。
- 10.0.0.2：由 AWS 保留。DNS 服务器的 IP 地址是 VPC 网络范围的基址 + 2。对于包含多个 CIDR 块的 VPC，DNS 服务器的 IP 地址位于主要 CIDR 中。我们还为 VPC 中的所有 CIDR 块预留了每个子网范围加二的基址。有关更多信息，请参阅 Amazon DNS 服务器。
- 10.0.0.3：由 AWS 保留，以供将来使用。
- 10.0.0.255：网络广播地址。我们在 VPC 中不支持广播，因此我们会保留此地址。

如果您使用命令行工具或 Amazon EC2 API 创建子网，则系统会自动将 CIDR 块修改为其规范形式。例如，假设您为 CIDR 块指定 100.68.0.18/18，我们将创建一个 CIDR 块 100.68.0.0/18。

使用 BYOIP 将 IPv4 地址范围设置为 AWS 后，您可以使用该范围内的所有 IP 地址，包括第一个地址（网络地址）和最后一个地址（广播地址）。

IPv6 的子网定型

如果您已向 VPC 关联 IPv6 CIDR 块，则可以将 IPv6 CIDR 块与 VPC 中的现有子网关联，或在创建新子网时执行此操作。可能的 IPv6 网络掩码长度介于 /44 和 /64 之间，增量为 /4。

互联网上提供的工具可帮助您计算和创建 IPv6 子网 CIDR 块。您可以通过搜索“IPv6 子网计算器”或“IPv6 CIDR 计算器”等术语来查找满足您需求的工具。您的网络工程组也可以帮助您判断可为您的子网指定哪些具体 IPv6 CIDR 块。

每个子网 CIDR 块中的前四个 IPv6 地址和最后一个 IPv6 地址无法供您使用，而且无法分配给 EC2 实例。例如，在具有 CIDR 块 2001:db8:1234:1a00/64 的子网中，以下五个 IP 地址是保留的：

- 2001:db8:1234:1a00::
- 2001:db8:1234:1a00::1：由 AWS 保留，用于 VPC 路由器。
- 2001:db8:1234:1a00::2
- 2001:db8:1234:1a00::3

- 2001:db8:1234:1a00:ffff:ffff:ffff:ffff

除了上述示例中 AWS 为 VPC 路由器预留的 IP 地址外，还为默认 VPC 路由器预留了以下 IPv6 地址：

- 使用 EUI-64 生成的位于 FE80::/10 范围内的链路本地 IPv6 地址。有关链路本地地址的更多信息，请参阅链路本地地址。
- 链路本地 IPv6 地址 FE80:ec2::1。

如果您需要通过 IPv6 与 VPC 路由器通信，您可以将应用程序配置为与最适合您需求的地址通信。

比较 IPv4 与 IPv6

下表总结 Amazon EC2 和 Amazon VPC 中 IPv4 与 IPv6 之间的差异。

有关支持双堆栈配置（IPv4 和 IPv6）和仅 IPv6 配置的 AWS 服务列表，请参阅 支持 IPv6 的服务。

特征	IPv4	IPv6
VPC 大小	最多 5 个 CIDR，从 /16 到 /28。此配额可调整。	最多 5 个 CIDR，从 /44 到 /60，以 /4 为增量。此配额可调整。
子网大小	从 /16 到 /28	从 /44 到 /64，增量为 /4。
地址选择	您可以选择 VPC 的 IPv4 CIDR 块，也可从 Amazon VPC IP 地址管理器（IPAM）中分配一个 CIDR 块。有关更多信息，请参阅《Amazon VPC IPAM 用户指南》中的什么是 IPAM？。	您可以将自己的 IPv6 CIDR 块带入 AWS 以用于您的 VPC，选择 Amazon 提供的 IPv6 CIDR 块，或者从 Amazon VPC IP 地址管理器（IPAM）中分配一个 CIDR 块。有关更多信息，请参阅《Amazon VPC IPAM 用户指南》中的什么是 IPAM？。
互联网访问	需要互联网网关。	需要互联网网关。支持使用仅限出口的互联网网关进行仅出站通信。
弹性 IP 地址	支持。为 EC2 实例提供永久的静态公有 IPv4 地址。	不支持。实例重启时，EIP 会使实例的公有 IPv4 地址保持静态。默认情况下 IPv6 地址是静态的。

特征	IPv4	IPv6
NAT 网关	支持。私有子网中的实例可以通过公有 NAT 网关连接到互联网，也可以使用私有 NAT 网关连接到其他 VPC 中的资源。	支持。您可以使用带有 NAT64 的 NAT 网关，使仅 IPv6 子网中的实例在 VPC 中、VPC 之间、本地网络中，或通过互联网仅与 IPv4 资源通信。
DNS 名称	实例将接收 Amazon 提供的 IPBN 或基于 RBN 的 DNS 名称。DNS 名称将解析为为实例选择的 DNS 记录。	实例将接收 Amazon 提供的 IPBN 或基于 RBN 的 DNS 名称。DNS 名称将解析为为实例选择的 DNS 记录。

使用托管前缀列表合并和管理网络 CIDR 块

托管式前缀列表是包含一个或多个 CIDR 块的集合。您可以使用前缀列表更轻松地配置和维护安全组和路由表。您可以根据经常使用的 IP 地址创建前缀列表，并将它们作为安全组规则和路由中的集合引用，而不是单独引用它们。例如，您可以将具有不同 CIDR 块但使用相同端口和协议的安全组规则整合到使用前缀列表的单个规则中。如果您扩展网络并需要允许来自另一个 CIDR 块的流量，则可以更新相关的前缀列表，使用该前缀列表的所有安全组都将更新。您还可以使用资源访问管理器（RAM）与其他 AWS 账户一起使用托管式前缀列表。

前缀列表有两种类型：

- 客户管理的前缀列表 — 您定义和管理的 IP 地址范围集。您可以与其他AWS账户共享您的前缀列表，使这些账户能够在自己的资源中引用该前缀列表。
- AWS托管前缀列表 — AWS服务的 IP 地址范围集。您无法创建、修改、共享或删除AWS托管的前缀列表。

目录

- 前缀列表概念和规则
- 适用于前缀列表的 Identity and Access Management
- 客户管理的前缀列表
- AWS托管前缀列表
- 使用前缀列表优化 AWS 基础设施管理

前缀列表概念和规则

前缀列表由条目 组成。每个条目均包含一个 CIDR 块和（可选）该 CIDR 块的描述。

客户管理的前缀列表

以下规则适用于客户管理的前缀列表：

- 前缀列表仅支持单一类型的 IP 寻址（IPv4 或 IPv6）。不能在单个前缀列表中组合 IPv4 和 IPv6 CIDR 块。
- 前缀列表仅适用于您创建它时所在的区域。
- 创建前缀列表时，必须指定前缀列表可支持的最大条目数。
- 当您在资源中引用前缀列表时，前缀列表的最大条目数占用资源的条目数限额。例如，如果您创建一个包含最多 20 个条目的前缀列表，并且在安全组规则中引用该前缀列表，这将视为 20 个安全组规则。
- 在路由表中引用前缀列表时，路由优先级规则适用。有关更多信息，请参阅 前缀列表的路由优先级。
- 可修改前缀列表。您添加或删除条目时，我们会创建新版本的前缀列表。引用前缀的资源始终使用当前（最新）版本。您可以从前缀列表的以前版本还原条目，这同样会创建新版本。
- 存在与前缀列表相关的配额。有关更多信息，请参阅 客户管理的前缀列表。
- 客户托管式前缀列表可在所有商业 AWS 区域中使用，包括 GovCloud（美国）和中国区域。

AWS托管前缀列表

以下规则适用于AWS托管的前缀列表：

- 您无法创建、修改、共享或删除AWS托管的前缀列表。
- 不同的 AWS 托管前缀列表在使用时具有不同的权重。有关更多信息，请参阅 AWS 托管前缀列表权重。
- 您无法查看AWS托管的前缀列表的版本号。

适用于前缀列表的 Identity and Access Management

默认情况下，用户无权创建、查看、修改或删除前缀列表。您可以创建一个 IAM 策略并附加一个允许用户使用前缀列表的角色。

要查看 Amazon VPC 操作以及您可以在 IAM 策略中使用的资源和条件键的列表，请参阅《服务授权参考》中的 Actions, resources, and condition keys for Amazon EC2。

以下示例策略仅允许用户查看和使用前缀列表 pl-123456abcde123456。用户无法创建或删除前缀列表。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:GetManagedPrefixListAssociations",
                "ec2:GetManagedPrefixListEntries",
                "ec2:ModifyManagedPrefixList",
                "ec2:RestoreManagedPrefixListVersion"
            ],
            "Resource": "arn:aws:ec2:us-east-1:123456789012:prefix-
list/pl-123456abcde123456"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeManagedPrefixLists",
            "Resource": "*"
        }
    ]
}
```

有关在 Amazon VPC 中使用 IAM 的更多信息，请参阅适用于 Amazon VPC 的 Identity and Access Management。

客户管理的前缀列表

您可以借助客户管理的前缀列表在 AWS 中定义并维护一组自己的 IP 地址范围（称为前缀）。您可以创建集中式前缀列表在需要时加以引用，不必将这些 IP 地址硬编码到各种资源中。这不仅简化了 IP 地址的管理，还提高了整个 AWS 环境的一致性和可重用性。

客户管理的前缀列表的一个突出功能是能够与其他 AWS 账户共享列表。通过授予对前缀列表的访问权限，您可以让其他团队或组织在他们自己的资源中利用您定义的 IP 地址范围。这种协作方法可营造更富凝聚力、更高效，能够共享和同步 IP 地址管理的云体验。

在接下来的各节中，我们将从实践层面更深入地探讨客户管理的前缀列表的使用，包括有关创建、管理和共享 IP 地址范围的分步指导。

任务

- 使用客户管理的前缀列表

使用客户管理的前缀列表

本节旨在介绍如何使用客户管理的前缀列表。

内容

- 创建前缀列表
- 查看前缀列表
- 查看前缀列表的条目
- 查看前缀列表的关联（引用）
- 修改前缀列表
- 调整前缀列表的大小
- 还原前缀列表的以前版本
- 删除前缀列表
- 共享客户管理的前缀列表

创建前缀列表

创建前缀列表时，必须指定前缀列表可支持的最大条目数。

限制

如果规则数加上前缀列表的最大条目数超过账户每个安全组的规则配额，则无法向安全组规则添加前缀列表。

使用控制台创建前缀列表

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。

2. 在导航窗格中，选择托管前缀列表。
3. 选择创建前缀列表。
4. 对于前缀列表名称，输入前缀列表的名称。
5. 对于最大条目数，输入前缀列表的最大条目数。
6. 对于地址系列，选择前缀列表是支持 IPv4 条目还是 IPv6 条目。
7. 对于前缀列表条目，选择添加新条目，然后输入 CIDR 块和条目的描述。对每个条目重复此步骤。
8. （可选）对于标签，将标签添加到前缀列表，以帮助您以后识别它。
9. 选择创建前缀列表。

使用 AWS CLI 创建前缀列表

使用 create-managed-prefix-list 命令。

查看前缀列表

您可以查看您的前缀列表、与您共享的前缀列表以及AWS托管的前缀列表。

使用控制台查看前缀列表

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择托管前缀列表。
3. 拥有者 ID 列显示前缀列表拥有者的 AWS 账户 ID。对于AWS托管前缀列表，Owner ID (拥有者 ID) 是AWS。

使用 AWS CLI 查看前缀列表

使用 describe-managed-prefix-lists 命令。

查看前缀列表的条目

您可以查看您的前缀列表的条目、与您共享的前缀列表以及AWS托管的前缀列表。

使用控制台查看前缀列表的条目

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择托管前缀列表。
3. 选中与所需前缀列表对应的复选框。

4. 在下部窗格中，选择条目以查看前缀列表的条目。

使用 AWS CLI 查看前缀列表的条目

使用 get-managed-prefix-list-entries 命令。

查看前缀列表的关联（引用）

您可以查看与前缀列表关联的资源的 ID 和拥有者。关联的资源是指在其条目或规则中引用前缀列表的资源。

限制

您无法查看AWS托管的前缀列表的关联资源。

使用控制台查看前缀列表关联

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择托管前缀列表。
3. 选中与所需前缀列表对应的复选框。
4. 在下部窗格中，选择关联以查看引用前缀列表的资源。

使用 AWS CLI 查看前缀列表关联

使用 get-managed-prefix-list-associations 命令。

修改前缀列表

您可以修改前缀列表的名称，也可以添加或删除条目。要修改最大条目数，请参阅 调整前缀列表的大小。

若更新前缀列表条目，系统会为前缀列表创建新版本。若更新前缀列表条目名称或条目上限，系统不会为前缀列表创建新版本。

注意事项

- 您不能修改AWS托管的前缀列表。
- 若增加前缀列表条目上限，增量会应用到引用此前缀列表的资源条目配额。若其中有任何资源不支持此上限增加，修改操作会失败，且并上限会恢复到之前大小。

使用控制台修改前缀列表

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择托管前缀列表。
3. 选中前缀列表的复选框，然后依次选择 Actions (操作)、Modify prefix list (修改前缀列表)。
4. 对于前缀列表名称，输入前缀列表的新名称。
5. 对于前缀列表条目，选择删除以删除现有条目。要添加新条目，请选择添加新条目，然后输入 CIDR 块和条目的描述。
6. 选择保存前缀列表。

使用 AWS CLI 修改前缀列表

使用 modify-managed-prefix-list 命令。

调整前缀列表的大小

您可以调整前缀列表的大小，并可以将前缀列表的最大条目数修改为 1000。有关客户托管的前缀列表配额的更多信息，请参阅 客户管理的前缀列表。

使用控制台调整前缀列表的大小

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择托管前缀列表。
3. 选中前缀列表的复选框，然后依次选择 Actions（操作）、Resize prefix list（调整前缀列表的大小）。
4. 对于 New max entries（新的最大条目数），请输入一个值。
5. 选择 Resize（调整大小）。

使用 AWS CLI 调整前缀列表的大小

使用 modify-managed-prefix-list 命令。

还原前缀列表的以前版本

您可以将条目从前缀列表的以前版本还原。这将创建前缀列表的一个新版本。

若减小前缀列表，必须确保前缀列表足以包含旧版本条目。

使用控制台还原前缀列表的以前版本

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择托管前缀列表。
3. 选中前缀列表的复选框，然后依次选择 Actions (操作)、Restore prefix list (还原前缀列表)。
4. 为 Select prefix list version (选择前缀列表版本) 选择旧版本。所选版本条目会在 Prefix list entries (前缀列表条目) 内显示。
5. 选择还原前缀列表。

使用 AWS CLI 还原前缀列表的以前版本

使用 restore-managed-prefix-list-version 命令。

删除前缀列表

要删除前缀列表，必须首先删除在资源中（例如在路由表中）对该列表的所有引用。如果您已使用 AWS RAM 共享前缀列表，则必须首先删除使用者拥有的资源中的所有引用。

限制

您不能删除AWS托管的前缀列表。

使用控制台删除前缀列表

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择托管前缀列表。
3. 选择前缀列表，然后依次选择操作、删除前缀列表。
4. 在确认对话框中，输入 `delete`，然后选择删除。

使用 AWS CLI 删除前缀列表

使用 delete-managed-prefix-list 命令。

共享客户管理的前缀列表

借助 AWS Resource Access Manager（AWS RAM），客户托管前缀列表的拥有者可以与以下对象共享前缀列表：

- AWS Organizations 中的拥有者企业内部或外部的特定 AWS 账户

- 中的所有者企业内部的企业单位AWS Organizations
- AWS Organizations 中的整个所织

已与之共享前缀列表的使用者可以查看前缀列表及其条目，也可以在其AWS资源中引用前缀列表。

有关 AWS RAM 的更多信息，请参阅 AWS RAM 用户指南。有关配额的更多信息，请参阅《AWS RAM User Guide》中的 Service quotas。

> **Important**
>
> 共享前缀列表不会产生额外的费用。

内容

- 共享前缀列表权限
- 使用共享前缀列表

共享前缀列表权限

拥有者的权限

拥有者负责管理共享前缀列表及其条目。拥有者可以查看引用前缀列表的AWS资源的 ID。但是，他们不能在使用者拥有的AWS资源中添加或删除对前缀列表的引用。

如果在使用者拥有的资源中引用了前缀列表，则拥有者不能删除该前缀列表。

使用者的权限

使用者可以查看共享前缀列表中的条目，也可以在其AWS资源中引用共享前缀列表。但是，使用者无法修改、还原或删除共享前缀列表。

使用共享前缀列表

AWS 前缀列表提供了一种便捷的方式来管理和引用各种 AWS 服务使用的 IP 地址范围。除了 AWS 托管前缀列表，您还可以创建自己的客户管理的前缀列表并与其他 AWS 账户共享。

共享前缀列表对于具有复杂联网要求的组织或需要在多个 AWS 工作负载之间协调 IP 地址使用的组织特别有用。通过共享前缀列表，您可以确保一致的 IP 地址管理，简化协作者的联网配置。

本节旨在介绍如何共享前缀列表，以及如何识别和使用已与自己账户共享的前缀列表。

内容

- 共享前缀列表
- 将共享前缀列表取消共享
- 识别共享前缀列表
- 识别对共享前缀列表的引用

共享前缀列表

要共享前缀列表，您必须将它添加到资源共享。如果您没有资源共享，则必须首先使用 AWS RAM 控制台创建一个。

如果您是 AWS Organizations 中某企业的一部分并且已在您的企业中启用共享，企业中的使用者将自动获得对共享前缀列表的访问权限。否则，使用者将会收到加入资源共享的邀请，并在接受邀请后获得对共享前缀列表的访问权限。

您可以使用 AWS RAM 控制台或 AWS CLI 创建资源共享并共享您拥有的前缀列表。

> **Important**
>
> - 要共享前缀列表，您必须拥有它。您无法共享已与您共享的前缀列表。您不能共享AWS托管的前缀列表。
> - 要与您的企业或 AWS Organizations 内的企业部门共享前缀列表，您必须允许与 AWS Organizations 共享。有关更多信息，请参阅《AWS RAM 用户指南》中的允许与 AWS Organizations 共享。

使用 AWS RAM 控制台创建资源共享并共享前缀列表

按照 AWS RAM 用户指南中创建资源共享的步骤操作。对于选择资源类型，选择前缀列表，然后选中您的前缀列表的复选框。

使用 AWS RAM 控制台将前缀列表添加到现有资源共享

要将您拥有的托管前缀添加到现有资源共享，请按照 AWS RAM 用户指南中更新资源共享的步骤操作。对于选择资源类型，选择前缀列表，然后选中您的前缀列表的复选框。

使用 AWS CLI 共享您拥有的前缀列表

使用以下命令创建和更新资源共享：

- create-resource-share
- associate-resource-share
- update-resource-share

将共享前缀列表取消共享

取消共享前缀列表后，使用者不再可以在其账户中查看前缀列表或条目，也无法在其资源中引用前缀列表。如果已在使用者的资源中引用前缀列表，则这些引用将继续正常运行，并且您可以继续查看这些引用。如果将前缀列表更新为新版本，则引用将使用最新版本。

要取消共享您拥有的已共享前缀列表，必须使用 AWS RAM 从资源共享中将其删除。

使用 AWS RAM 控制台取消共享您拥有的共享前缀列表

请参阅 AWS RAM 用户指南中的更新资源共享。

使用 AWS CLI 取消共享您拥有的共享前缀列表

使用 disassociate-resource-share 命令。

识别共享前缀列表

拥有者和使用者可以使用 Amazon VPC 控制台和 AWS CLI 识别共享前缀列表。

使用 Amazon VPC 控制台识别共享前缀列表

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择托管前缀列表。
3. 此页面显示您拥有的前缀列表以及与您共享的前缀列表。拥有者 ID 列显示前缀列表拥有者的 AWS 账户 ID。
4. 要查看前缀列表的资源共享信息，请选择该前缀列表，然后选择下部窗格中的共享。

使用 AWS CLI 识别共享的前缀列表

使用 describe-managed-prefix-lists 命令。该命令返回您拥有的前缀列表以及与您共享的前缀列表。`OwnerId` 显示前缀列表拥有者的 AWS 账户 ID。

识别对共享前缀列表的引用

所有者可以识别使用者拥有的引用共享前缀列表的资源。

使用 Amazon VPC 控制台识别对共享前缀列表的引用

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择托管前缀列表。
3. 选择前缀列表，然后选择下部窗格中的关联。
4. 引用前缀列表的资源的 ID 列在资源 ID 列中。资源的拥有者列在资源拥有者列中。

使用 AWS CLI 识别对共享前缀列表的引用

使用 get-managed-prefix-list-associations 命令。

AWS托管前缀列表

AWS 托管前缀列表是 AWS 服务的 IP 地址范围集。这些前缀列表由亚马逊云科技维护，提供了引用各种 AWS 服务所用 IP 地址的方法。在 VPC 中配置安全组或其他网络级控制时，此功能可能特别有用。

前缀列表适用于广泛的 AWS 服务，包括 S3 和 DynamoDB。通过使用托管前缀列表，您可以确保自己的网络配置处于最新状态，并且可以正确考虑自己依赖的 AWS 服务所使用的 IP 地址。这有助于简化联网任务，减少手动维护 IP 地址列表的管理开销。

除了实际优势外，使用托管前缀列表还符合 AWS 安全最佳实践。依靠 AWS 提供的权威 IP 地址信息，您可以最大限度地降低配置错误或意外连接问题的风险。这对于具有严格合规性要求的任务关键型应用程序或工作负载尤其重要。

内容

- 可用的 AWS 托管前缀列表
- AWS 托管前缀列表权重
- 使用 AWS 托管前缀列表

可用的 AWS 托管前缀列表

以下服务提供 AWS 托管前缀列表。

AWS 服务	前缀列表名称	权重
Amazon CloudFront	com.amazonaws.global.cloudfront.origin-facing	55
Amazon DynamoDB	com.amazonaws.*region*.dynamodb	1

AWS 服务	前缀列表名称	权重
Amazon EC2 Instance Connect	com.amazonaws.*region*.ec2-instance-connect	2
	com.amazonaws.*region*.ipv6.ec2-instance-connect	2
AWS Ground Station	com.amazonaws.global.groundstation	5
Amazon Route 53	com.amazonaws.*region*.ipv6.route53-healthchecks	25
	com.amazonaws.*region*.route53-healthchecks	25
Amazon S3	com.amazonaws.*region*.s3	1
Amazon S3 Express One Zone 存储类	com.amazonaws.*region*.s3express	6
Amazon VPC Lattice	com.amazonaws.*region*.vpc-lattice	10
	com.amazonaws.*region*.ipv6.vpc-lattice	10

使用控制台查看 AWS 托管前缀列表

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择托管前缀列表。
3. 在搜索字段中，添加拥有者 ID：AWS 筛选条件。

使用 AWS CLI 查看 AWS 托管前缀列表

使用 describe-managed-prefix-lists 命令，如下所示。

```
aws ec2 describe-managed-prefix-lists --filters Name=owner-id,Values=AWS
```

AWS 托管前缀列表权重

AWS 托管前缀列表权重是指前缀列表将在资源中占用的条目数。

例如，Amazon CloudFront 托管前缀列表的权重为 55。以下是将对 Amazon VPC 配额产生的影响：

- 在安全组中，默认配额为 60 条规则，在一个安全组中仅保留 5 条额外规则的空间。对于此配额，您可以请求增加配额。
- 在路由表中，默认配额为 50 个路由，因此您必须请求增加配额才能将前缀列表添加到路由表。

使用 AWS 托管前缀列表

AWS 托管前缀列表由 AWS 创建和维护，并且可供任何拥有 AWS 账户的人员使用。您无法创建、修改、共享或删除AWS托管的前缀列表。

与客户管理的前缀列表一样，您可以将 AWS 托管前缀列表与安全组和路由表等 AWS 资源结合使用。有关更多信息，请参阅 使用前缀列表优化 AWS 基础设施管理。

使用前缀列表优化 AWS 基础设施管理

您可以在以下AWS资源中引用前缀列表。

资源

- VPC 安全组
- 子网路由表
- 中转网关路由表
- AWS Network Firewall 规则组
- Amazon Managed Grafana 网络访问控制
- AWS Outposts 机架本地网关

VPC 安全组

您可以将前缀列表指定为入站规则的源或出站规则的目的地。有关更多信息，请参阅 安全组。

> **Important**
>
> 您无法修改现有规则来使用前缀列表。而须创建新规则才能使用前缀列表。

使用控制台在安全组规则中引用前缀列表

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。

2. 在导航窗格中，选择 Security Groups。
3. 选择需要更新的安全组。
4. 依次选择 Actions (操作)、Edit inbound rules (编辑入站规则)，或 Actions (操作)、Edit outbound rules (编辑出站规则)。
5. 选择 Add rule。对于类型，选择流量类型。对于源（入站规则）或目的地（出站规则），选择“自定义”。然后，在“前缀列表”下的下一个字段中，选择前缀列表的 ID。
6. 选择 Save rules (保存规则)。

使用 AWS CLI 在安全组规则中引用前缀列表

使用 authorize-security-group-ingress 和 authorize-security-group-egress 命令。对于 `--ip-permissions` 参数，请使用 `PrefixListIds` 指定前缀列表的 ID。

子网路由表

您可以将前缀列表指定为路由表条目的目的地。不能在网关路由表中引用前缀列表。有关路由表的更多信息，请参见配置路由表。

使用控制台在路由表中引用前缀列表

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择路由表，然后选择路由表。
3. 依次选择 Actions (操作)、Edit routes (编辑路由)。
4. 要添加路由，请选择添加路由。
5. 对于目的地，输入前缀列表的 ID。
6. 对于目标，请选择一个目标。
7. 选择保存更改。

使用 AWS CLI 在路由表中引用前缀列表

使用 create-route (AWS CLI) 命令。使用 `--destination-prefix-list-id` 参数指定前缀列表的 ID。

中转网关路由表

您可以将前缀列表指定为路由目的地。有关更多信息，请参阅 Amazon VPC 中转网关 中的前缀列表参考。

AWS Network Firewall 规则组

AWS Network Firewall 规则组是检查和处理网络流量的一组可重复使用的标准。如果您在 AWS Network Firewall 中创建与 Suricata 兼容的有状态规则组，则可以引用规则组中的前缀列表。有关更多信息，请参阅 AWS Network Firewall 开发人员指南中的引用 Amazon VPC 前缀列表和创建有状态规则组。

Amazon Managed Grafana 网络访问控制

对于向 Amazon Managed Grafana 工作区发出的请求，您可以指定一个或多个前缀列表作为入站规则。有关 Grafana 工作区网络访问控制的更多信息（包括如何引用前缀列表），请参阅《Amazon Managed Grafana 用户指南》中的管理网络访问。

AWS Outposts 机架本地网关

每个 AWS Outposts 机架都提供一个本地网关，允许您将 Outpost 资源与本地网络连接起来。您可以将经常使用的 CIDR 分组到前缀列表中，并引用该列表作为本地网关路由表中的路由目标。有关更多信息，请参阅《AWS Outposts 机架用户指南》中的管理本地网关路由表路由。

AWS IP 地址范围

AWS 以 JSON 格式发布其当前的 IP 地址范围。利用这些信息，您可以识别来自 AWS 的流量。这些信息也可用于允许或拒绝发往或来自某些 AWS 服务 的流量。

注意事项

- 我们会发布客户通常用于执行出口筛选的服务的 IP 地址范围。我们不会公布所有服务的 IP 地址范围。
- 服务可以使用其 IP 地址范围与其他服务通信，也可以使用这些 IP 范围与客户网络通信。
- 通过自带 IP 地址（BYOIP）引入到 AWS 的 IP 地址范围不包含在 `.json` 文件内。有关更多信息，请参阅《Amazon EC2 用户指南》中的通过 AWS 公告地址范围。

某些服务使用 AWS 托管式前缀列表发布其地址范围。有关更多信息，请参阅 the section called “可用的 AWS 托管前缀列表”。

内容

- 下载 JSON 文件

- 出口控制
- 地理位置源
- 查找 AWS 服务 的 IP 地址范围
- AWS IP 地址范围 JSON 的语法
- AWS IP 地址范围通知

下载 JSON 文件

要查看当前的地址范围，请下载 ip-ranges.json。要维护历史记录，请将连续版本的 JSON 文件保存在自己的计算机上。要确定自上次保存文件以来是否发生更改，请检查当前文件中的发布时间，并将其与上次保存文件中的发布时间进行比较。

以下是将 JSON 文件保存到当前目录的 curl 命令示例。

```
curl -O https://ip-ranges.amazonaws.com/ip-ranges.json
```

如果您以编程方式访问此文件，您有责任确保仅在成功验证服务器提供的 TLS 证书之后，应用程序才能下载文件。

要接收 JSON 文件更新通知，请参阅AWS IP 地址范围通知。

出口控制

要允许使用一项 AWS 服务创建的资源仅访问其他 AWS 服务，可以使用 ip-ranges.json 文件中的 IP 地址范围信息来执行出口筛选。确保安全组规则允许出站流量流向 AMAZON 列表中的 CIDR 块。安全组存在限额。根据每个区域中 IP 地址范围的数量，每个区域可能需要使用多个安全组。

> **Note**
>
> 有些 AWS 服务基于 EC2 构建并使用 EC2 IP 地址空间。如果您屏蔽流向 EC2 IP 地址空间的流量，则也将阻止这些非 EC2 服务的流量。

地理位置源

`ip-ranges.json` 的 IP 地址范围按照 AWS 区域。但是，本地区域与其父区域不在同个物理位置。geo-ip-feed.csv 中发布的地理位置数据考虑了本地区域。数据遵循 RFC 8805。

查找 AWS 服务 的 IP 地址范围

AWS 提供的 AWS IP 地址范围 JSON 文件可以作为查找各种 AWS 服务的 IP 地址并利用该信息增强网络安全和访问控制的宝贵资源。通过解析该 JSON 文件中包含的详细数据，即可精确识别与特定 AWS 服务 和区域关联的 IP 地址范围。

例如，您可以利用 IP 地址范围来配置强大的网络安全策略，设置精细的防火墙规则以允许或拒绝对某些 AWS 资源的访问。此信息也适用于各种 AWS Network Firewall 任务。这种控制级别对保护应用程序和数据至关重要，可确保只有经过授权的流量才能到达必要的 AWS 服务。此外，拥有这种 IP 情报有助于正确配置应用程序以与正确的 AWS 端点通信，从而提高整体可靠性和性能。

除了防火墙规则，还可以利用 `ip-ranges.json` 文件在网络基础设施上配置复杂的出口筛选。通过了解不同 AWS 服务 的目标 IP 地址范围，您可以设置路由策略或利用类似的高级网络安全解决方案，从而根据预期目标有选择地允许或阻止出站流量。这种出口控制对于降低数据泄露和未经授权访问的风险至关重要。

值得注意的是，`ip-ranges.json` 文件会定期更新，因此维护最新的本地副本对于确保您获得最准确且最新的信息至关重要。通过持续利用此文件的内容，您可以有效地管理基于 AWS 的应用程序的网络访问和安全性，从而增强您的整体云安全状况。

以下示例有助于筛选 AWS IP 地址范围，以便找出符合需求的 IP 地址。在 Linux 上，您可以下载并使用 jq 工具来解析 JSON 文件的本地副本。AWS Tools for Windows PowerShell 包含的 cmdlet（即 Get-AWSPublicIpAddressRange）可用于解析该 JSON 文件。有关更多信息，请参阅以下博客文章：《Querying the Public IP Address Ranges for AWS》。

要获取 JSON 文件，请参阅the section called “下载”。有关 JSON 文件语法的更多信息，请参阅 the section called “语法”。

示例

- 获取文件创建日期
- 获取指定区域的 IP 地址
- 获取所有 IPv4 地址
- 获取特定服务的所有 IPv4 地址
- 获取特定区域中的特定服务的所有 IPv4 地址
- 获取所有 IPv6 地址
- 获取特定服务的所有 IPv6 地址

- 获取指定边界组的所有 IP 地址

获取文件创建日期

下面的示例获取了 ip-ranges.json 的创建日期。

jq

```
$ jq .createDate < ip-ranges.json

"2024-08-01-17-22-15"
```

PowerShell

```
PS C:\> Get-AWSPublicIpAddressRange -OutputPublicationDate

Thursday, August 1, 2024 9:22:35 PM
```

获取指定区域的 IP 地址

以下示例筛选了 JSON 文件中指定区域的 IP 地址。

jq

```
$ jq '.prefixes[] | select(.region=="us-east-1")' < ip-ranges.json

{
  "ip_prefix": "23.20.0.0/14",
  "region": "us-east-1",
  "network_border_group": "us-east-1",
  "service": "AMAZON"
},
{
  "ip_prefix": "50.16.0.0/15",
  "region": "us-east-1",
  "network_border_group": "us-east-1",
  "service": "AMAZON"
},
{
  "ip_prefix": "50.19.0.0/16",
```

```
  "region": "us-east-1",
  "network_border_group": "us-east-1",
  "service": "AMAZON"
},
...
```

PowerShell

```
PS C:\> Get-AWSPublicIpAddressRange -Region us-east-1

IpPrefix        Region      NetworkBorderGroup         Service
--------        ------       -------                   -------
23.20.0.0/14    us-east-1    us-east-1                 AMAZON
50.16.0.0/15    us-east-1    us-east-1                 AMAZON
50.19.0.0/16    us-east-1    us-east-1                 AMAZON
...
```

获取所有 IPv4 地址

以下示例筛选了 JSON 文件中的 IPv4 地址。

jq

```
$ jq -r '.prefixes | .[].ip_prefix' < ip-ranges.json

23.20.0.0/14
27.0.0.0/22
43.250.192.0/24
...
```

PowerShell

```
PS C:\> Get-AWSPublicIpAddressRange | where {$_.IpAddressFormat -eq "Ipv4"} | select
 IpPrefix

IpPrefix
--------
23.20.0.0/14
27.0.0.0/22
43.250.192.0/24
...
```

获取特定服务的所有 IPv4 地址

以下示例筛选了 JSON 文件中指定服务的 IPv4 地址。

jq

```
$ jq -r '.prefixes[] | select(.service=="GLOBALACCELERATOR") | .ip_prefix' < ip-
ranges.json

13.248.117.0/24
15.197.34.0/23
15.197.36.0/22
...
```

PowerShell

```
PS C:\> Get-AWSPublicIpAddressRange -ServiceKey GLOBALACCELERATOR | where
 {$_.IpAddressFormat -eq "Ipv4"} | select IpPrefix

IpPrefix
--------
13.248.117.0/24
15.197.34.0/23
15.197.36.0/22
...
```

获取特定区域中的特定服务的所有 IPv4 地址

以下示例筛选了 JSON 文件中指定区域内指定服务的 IPv4 地址。

jq

```
$ jq -r '.prefixes[] | select(.region=="us-east-1") |
 select(.service=="GLOBALACCELERATOR") | .ip_prefix' < ip-ranges.json

13.248.124.0/24
99.82.166.0/24
99.82.171.0/24
...
```

PowerShell

```
PS C:\> Get-AWSPublicIpAddressRange -Region us-east-1 -ServiceKey GLOBALACCELERATOR
 | where {$_.IpAddressFormat -eq "Ipv4"} | select IpPrefix

IpPrefix
--------
13.248.117.0/24
99.82.166.0/24
99.82.171.0/24
...
```

获取所有 IPv6 地址

以下示例筛选了 JSON 文件中的 IPv6 地址。

jq

```
$ jq -r '.ipv6_prefixes | .[].ipv6_prefix' < ip-ranges.json

2a05:d07c:2000::/40
2a05:d000:8000::/40
2406:dafe:2000::/40
...
```

PowerShell

```
PS C:\> Get-AWSPublicIpAddressRange | where {$_.IpAddressFormat -eq "Ipv6"} | select
 IpPrefix

IpPrefix
--------
2a05:d07c:2000::/40
2a05:d000:8000::/40
2406:dafe:2000::/40
...
```

获取特定服务的所有 IPv6 地址

以下示例筛选了 JSON 文件中指定服务的 IPv6 地址。

jq

```
$ jq -r '.ipv6_prefixes[] | select(.service=="GLOBALACCELERATOR") | .ipv6_prefix' <
 ip-ranges.json

2600:1f01:4874::/47
2600:1f01:4802::/47
2600:1f01:4860::/47
2600:9000:a800::/40
...
```

PowerShell

```
PS C:\> Get-AWSPublicIpAddressRange -ServiceKey GLOBALACCELERATOR | where
 {$_.IpAddressFormat -eq "Ipv6"} | select IpPrefix

IpPrefix
--------
2600:1f01:4874::/47
2600:1f01:4802::/47
2600:1f01:4860::/47
2600:9000:a800::/40
...
```

获取指定边界组的所有 IP 地址

以下示例筛选了 JSON 文件中指定边界组的所有 IP 地址。

jq

```
$ jq -r '.prefixes[] | select(.network_border_group=="us-west-2-lax-1")
 | .ip_prefix' < ip-ranges.json
70.224.192.0/18
52.95.230.0/24
15.253.0.0/16
...
```

PowerShell

```
PS C:\> Get-AWSPublicIpAddressRange | where {$_.NetworkBorderGroup -eq "us-west-2-
lax-1"} | select IpPrefix
```

```
IpPrefix
--------
70.224.192.0/18
52.95.230.0/24
15.253.0.0/16
...
```

AWS IP 地址范围 JSON 的语法

AWS 以 JSON 格式发布其当前的 IP 地址范围。要获取 JSON 文件，请参阅the section called “下载”。JSON 文件的语法如下。

```
{
  "syncToken": "0123456789",
  "createDate": "yyyy-mm-dd-hh-mm-ss",
  "prefixes": [
    {
      "ip_prefix": "cidr",
      "region": "region",
      "network_border_group": "network_border_group",
      "service": "subset"
    }
  ],
  "ipv6_prefixes": [
    {
      "ipv6_prefix": "cidr",
      "region": "region",
      "network_border_group": "network_border_group",
      "service": "subset"
    }
  ]
}
```

syncToken

采用 Unix 纪元时间格式的发布时间。

类型：字符串

示例："syncToken": "1416435608"

createDate

发布日期和时间，采用 UTC YY-MM-DD-hh-mm-ss 格式。

类型：字符串

示例：`"createDate": "2014-11-19-23-29-02"`

prefixes

IPv4 地址范围的 IP 前缀。

类型：数组

ipv6_prefixes

IPv6 地址范围的 IP 前缀。

类型：数组

ip_prefix

用 CIDR 表示法指定的公有 IPv4 地址范围。请注意，AWS 可在更具体的范围内公布前缀。例如，文件中的前缀 96.127.0.0/17 可公布为 96.127.0.0/21、96.127.8.0/21、96.127.32.0/19 和 96.127.64.0/18。

类型：字符串

示例：`"ip_prefix": "198.51.100.2/24"`

ipv6_prefix

用 CIDR 表示法指定的公有 IPv6 地址范围。请注意，AWS 可在更具体的范围内公布前缀。

类型：字符串

示例：`"ipv6_prefix": "2001:db8:1234::/64"`

network_border_group

网络边界组的名称，这是 AWS 通告 IP 地址或 GLOBAL 的可用区或本地区域的唯一集合。GLOBAL 服务流量可以被吸引到或来自 AWS 从中通告 IP 地址的多个（最多全部）可用区或本地区域。

类型：字符串

示例：`"network_border_group": "us-west-2-lax-1"`

区域

AWS 区域或 GLOBAL。GLOBAL 服务流量可以被吸引到或来自多个（最多全部）AWS 区域。

类型：字符串

有效值：af-south-1 | ap-east-1 | ap-east-2 | ap-northeast-1 | ap-northeast-2 | ap-northeast-3 | ap-south-1 | ap-south-2 | ap-southeast-1 | ap-southeast-2 | ap-southeast-3 | ap-southeast-4 | ap-southeast-5 | ap-southeast-7 | ca-central-1 | ca-west-1 | cn-north-1 | cn-northwest-1 | eu-central-1 | eu-central-2 | eu-north-1 | eu-south-1 | eu-south-2 | eu-west-1 | eu-west-2 | eu-west-3 | il-central-1 | mx-central-1 | me-central-1 | me-south-1 | sa-east-1 | us-east-1 | us-east-2 | us-gov-east-1 | us-gov-west-1 | us-west-1 | us-west-2 | GLOBAL

示例："region": "us-east-1"

service

IP 地址范围的子集。为 API_GATEWAY 列出的地址仅为出口 IP 地址。指定 AMAZON 可获得所有 IP 地址范围（这意味着每个子集也在 AMAZON 子集中）。但是，某些 IP 地址范围仅在 AMAZON 子集中（这意味着它们不会再包含在其他子集中）。

类型：字符串

有效值：AMAZON | AMAZON_APPFLOW | AMAZON_CONNECT | API_GATEWAY | AURORA_DSQL | CHIME_MEETINGS | CHIME_VOICECONNECTOR | CLOUD9 | CLOUDFRONT | CLOUDFRONT_ORIGIN_FACING | CODEBUILD | DYNAMODB | EBS | EC2 | EC2_INSTANCE_CONNECT | GLOBALACCELERATOR | IVS_LOW_LATENCY | IVS_REALTIME | KINESIS_VIDEO_STREAMS | MEDIA_PACKAGE_V2 | ROUTE53 | ROUTE53_HEALTHCHECKS | ROUTE53_HEALTHCHECKS_PUBLISHING | ROUTE53_RESOLVER | S3 | WORKSPACES_GATEWAYS

示例："service": "AMAZON"

范围重叠

任何服务代码返回的 IP 地址范围也由 AMAZON 服务代码返回。例如，由 S3 服务代码返回的所有 IP 地址范围也由 AMAZON 服务代码返回。

当服务 A 使用来自服务 B 的资源时，存在由服务 A 和服务 B 的服务代码返回的 IP 地址范围。然而，这些 IP 地址范围仅由服务 A 使用，而不能由服务 B 使用。例如，AmazonS3 使用来自 AmazonEC2 的资源，因此存在由 S3 和 EC2 服务代码返回的 IP 位置范围。但是，这些 IP 地址范围仅由 Amazon

S3 使用。因此，S3 服务代码会返回 Amazon S3 专门使用的所有 IP 地址范围。要识别 Amazon EC2 专门使用的 IP 地址范围，请查找 EC2 服务代码（而不是 S3 服务代码）返回的 IP 地址范围。

了解更多

本节旨在提供不同服务代码的附加信息链接。

- AMAZON_APPFLOW – IP 地址范围
- AMAZON_CONNECT – 设置您的网络
- CHIME_MEETINGS – 配置媒体和信号
- CLOUDFRONT – CloudFront 边缘服务器的位置和 IP 地址范围
- DYNAMODB – IP 地址范围
- EC2 – 公有 IPV4 地址
- EC2_INSTANCE_CONNECT – EC2 实例连接先决条件
- GLOBALACCELERATOR – Global Accelerator 边缘服务器的位置和 IP 地址范围
- ROUTE53 – Amazon Route 53 服务器的 IP 地址范围
- ROUTE53_HEALTHCHECKS – Amazon Route 53 服务器的 IP 地址范围
- ROUTE53_HEALTHCHECKS_PUBLISHING – Amazon Route 53 服务器的 IP 地址范围
- WORKSPACES_GATEWAYS–PCoIP 网关服务器

发行说明

下表介绍了 ip-ranges.json 语法的更新。我们还会在推出每个区域时添加新的区域代码。

描述	发行日期
添加了 IVS_LOW_LATENCY 服务代码。	2025 年 7 月 29 日
添加了 AURORA_DSQL 服务代码。	2025 年 5 月 21 日
添加了 IVS_REALTIME 服务代码。	2024 年 6 月 11 日
添加了 MEDIA_PACKAGE_V2 服务代码。	2023 年 5 月 9 日
添加了 CLOUDFRONT_ORIGIN_FACING 服务代码。	2021 年 10 月 12 日

描述	发行日期
添加了 ROUTE53_RESOLVER 服务代码。	2021 年 6 月 24 日
添加了 EBS 服务代码。	2021 年 5 月 12 日
添加了 KINESIS_VIDEO_STREAMS 服务代码。	2020 年 11 月 19 日
添加了 CHIME_MEETINGS 和 CHIME_VOICECONNECTOR 服务代码。	2020 年 6 月 19 日
添加了 AMAZON_APPFLOW 服务代码。	2020 年 6 月 9 日
增加了对网络边界组的支持。	2020 年 4 月 7 日
添加了 WORKSPACES_GATEWAYS 服务代码。	2020 年 3 月 30 日
添加了 ROUTE53_HEALTHCHECK_PUBLISHING 服务代码。	2020 年 1 月 30 日
添加了 API_GATEWAY 服务代码。	2019 年 9 月 26 日
添加了 EC2_INSTANCE_CONNECT 服务代码。	2019 年 6 月 26 日
添加了 DYNAMODB 服务代码。	2019 年 4 月 25 日
添加了 GLOBALACCELERATOR 服务代码。	2018 年 12 月 20 日
添加了 AMAZON_CONNECT 服务代码。	2018 年 6 月 20 日
添加了 CLOUD9 服务代码。	2018 年 6 月 20 日
添加了 CODEBUILD 服务代码。	2018 年 4 月 19 日
添加了 S3 服务代码。	2017 年 2 月 28 日
添加了对 IPv6 地址范围的支持。	2016 年 8 月 22 日
初始版本	2014 年 11 月 19 日

AWS IP 地址范围通知

AWS 以 JSON 格式发布其当前的 IP 地址范围。只要 AWS IP 地址范围发生更改，我们就会向名为 AmazonIpSpaceChanged 的 Amazon SNS 主题的订阅用户发送通知。有关 JSON 文件语法的更多信息，请参阅 the section called “语法”。

通知的有效负载包含以下格式的信息。

```
{
  "create-time":"yyyy-mm-ddThh:mm:ss+00:00",
  "synctoken":"0123456789",
  "md5":"6a45316e8bc9463c9e926d5d37836d33",
  "url":"https://ip-ranges.amazonaws.com/ip-ranges.json"
}
```

create-time

创建日期和时间。

通知可能不按顺序传输。因此，我们建议您检查时间戳以确保正确的顺序。

synctoken

采用 Unix 纪元时间格式的发布时间。

md5

ip-ranges.json 文件的加密哈希值。可以使用此值来检查下载的文件是否已损坏。

url

ip-ranges.json 文件的位置。有关更多信息，请参阅 the section called “下载”。

您可以按如下方式订阅接收通知。

订阅AWS IP 地址范围通知

1. 通过以下网址打开 Amazon SNS 控制台：https://console.aws.amazon.com/sns/v3/home。
2. 如果需要，可在导航栏中将区域更改为美国东部（弗吉尼亚北部）。您必须选择此区域，因为您订阅的 SNS 通知是在此区域中创建的。
3. 在导航窗格中，选择 Subscriptions。

4. 选择 Create subscription。
5. 在 Create subscription 对话框中，执行以下操作：
   a. 对于 Topic ARN，复制以下 Amazon Resource Name（ARN）：

      ```
      arn:aws:sns:us-east-1:806199016981:AmazonIpSpaceChanged
      ```

   b. 对于 Protocol，选择要使用的协议 (例如 `Email`)。
   c. 对于端点，键入用于接收通知的端点（例如，您的电子邮件地址）。
   d. 选择创建订阅。
6. 将通过您指定的端点与您联系并要求您确认订阅。例如，如果指定了电子邮件地址，您会收到一封主题行为 `AWS Notification - Subscription Confirmation` 的电子邮件。请按照说明确认订阅。

通知受到端点可用性约束。因此，应定期检查 JSON 文件以确保您在最新范围内。有关 Amazon SNS 可靠性的更多信息，请参阅https://aws.amazon.com/sns/faqs/#Reliability。

如果您不希望再收到这些通知，请通过以下步骤取消订阅。

取消订阅AWS IP 地址范围通知

1. 通过以下网址打开 Amazon SNS 控制台：https://console.aws.amazon.com/sns/v3/home。
2. 在导航窗格中，选择 Subscriptions。
3. 选中订阅对应的复选框。
4. 选择 Actions 和 Delete subscriptions。
5. 当系统提示进行确认时，选择 Delete（删除）。

有关 Amazon SNS 的更多信息，请参阅《Amazon Simple Notification Service 开发人员指南》。

VPC 的 IPv6 支持

如果您的现有 VPC 仅支持 IPv4 并且您的子网中的资源配置为仅使用 IPv4，则可为您的 VPC 和资源添加 IPv6 支持。您的 VPC 可在双堆栈模式下运行：您的资源可通过 IPv4 和/或 IPv6 进行通信。IPv4 和 IPv6 通信彼此独立。

您不能为 VPC 和子网禁用 IPv4 支持；这是 Amazon VPC 和 Amazon EC2 的默认 IP 寻址系统。

注意事项

- 不能从仅 IPv4 子网迁移到仅 IPv6 子网。
- 此示例假定您已有一个包含公有和私有子网的 VPC。有关创建新的 VPC 以用于 IPv6 的信息，请参阅 the section called “创建 VPC”。
- 开始使用 IPv6 之前，请确保您已了解 Amazon VPC 的 IPv6 寻址的功能：比较 IPv4 与 IPv6。

内容

- 为 VPC 添加 IPv6 支持
- 双堆栈 VPC 配置示例

为 VPC 添加 IPv6 支持

下表概述了为您的 VPC 启用 IPv6 的过程。

内容

- 步骤 1：将 IPv6 CIDR 块与您的 VPC 和子网关联
- 步骤 2：更新路由表
- 步骤 3：更新安全组规则
- 步骤 4：为实例分配 IPv6 地址

步骤	备注
步骤 1：将 IPv6 CIDR 块与您的 VPC 和子网关联	将 Amazon 提供的或 BYOIP IPv6 CIDR 块与您的 VPC 和子网关联。
步骤 2：更新路由表	更新路由表以路由 IPv6 流量。对于公有子网，请创建一个将所有 IPv6 流量都从该子网路由到互联网网关的路由。对于私有子网，请创建一个将所有发送到 Internet 的 IPv6 流量都从该子网路由到仅出口互联网网关的路由。
步骤 3：更新安全组规则	将安全组规则更新为包括 IPv6 地址规则。这样，使 IPv6 流量可以流入和流出您的实例。如

步骤	备注
	果您已创建自定义网络 ACL 规则来控制出入子网的流量，则必须包括 IPv6 流量规则。
步骤 4：为实例分配 IPv6 地址	将 IPv6 地址分配到您的子网的 IPv6 地址范围中的实例。

步骤 1：将 IPv6 CIDR 块与您的 VPC 和子网关联

您可将 IPv6 CIDR 块与 VPC 关联，然后将该范围内的一个 /64 CIDR 块与每个子网关联。

将 IPv6 CIDR 块与 VPC 关联

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Your VPCs(您的 VPC)。
3. 选择您的 VPC。
4. 选择操作、编辑 CIDR，然后选择添加新的 IPv6 CIDR。
5. 选择以下选项之一，然后选择选择 CIDR：
   - Amazon 提供的 IPv6 CIDR 块：使用 Amazon 的 IPv6 地址池中的 IPv6 CIDR 块。对于网络边界组，选择 AWS 将从中发布 IP 地址的组。
   - IPAM 分配的 IPv6 CIDR 块 – 使用 IPAM 池中的 IPv6 CIDR 块。选择 IPAM 池和 IPv6 CIDR 块。
   - 我拥有的 IPv6 CIDR – 使用您的 IPv6 地址池（BYOIP）中的 IPv6 CIDR 块。选择 IPv6 地址池和 IPv6 CIDR 块。
6. 选择关闭。

将 IPv6 CIDR 块与子网关联

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Subnets(子网)。
3. 选择子网。
4. 选择操作、编辑 IPv6 CIDR，然后选择添加 IPv6 CIDR。
5. 根据需要编辑 CIDR 块（例如，替换 00）。

6. 选择保存。
7. 对 VPC 中的任何其它子网重复此程序。

有关更多信息，请参阅 IPv6 VPC CIDR 块。

步骤 2：更新路由表

当您将 IPv6 CIDR 块与您的 VPC 关联时，我们会自动为该 VPC 的每个路由表添加本地路由，以允许 VPC 内的 IPv6 流量。

对于公有子网，您必须更新路由表，以使实例（例如 Web 服务器）能对 IPv6 流量使用互联网网关。对于私有子网，您必须更新路由表，以使实例（例如数据库实例）能对 IPv6 流量使用仅出口互联网网关，因为 NAT 网关不支持 IPv6。

要为公有子网更新路由表

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Subnets(子网)。选择公有子网。在路由表选项卡上，选择路由表 ID 以打开路由表的详细信息页面。
3. 选择 路由表。在 Routes (路由) 选项卡上，选择 Edit routes (编辑路由)。
4. 选择 Add route (添加路由)。对于目标，选择 `::/0`。选择适用于目标的互联网网关 ID。
5. 选择保存更改。

为私有子网更新路由表

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择仅出口互联网网关。选择创建仅出口互联网网关。从 VPC 中选择您的 VPC，然后选择创建仅出口互联网网关。

   有关更多信息，请参阅 使用仅出口互联网网关允许出站 IPv6 流量。
3. 在导航窗格中，选择 Subnets(子网)。选择私有子网。在路由表选项卡上，选择路由表 ID 以打开路由表的详细信息页面。
4. 选择 路由表。在 Routes (路由) 选项卡上，选择 Edit routes (编辑路由)。
5. 选择 Add route (添加路由)。对于目标，选择 `::/0`。请为目标选择仅出口互联网网关的 ID。
6. 选择保存更改。

> **Note**
>
> 路由表不能具有同时指向互联网网关和仅出口互联网网关的同一目的地 (::/0)。如果在配置仅出口互联网网关时收到一条错误消息，指出“存在以互联网网关为下一跃点的现有 IPv6 路由”，则必须先移除到互联网网关的现有 IPv6 路由，然后再将该路由添加到仅出口互联网网关。

有关更多信息，请参阅 示例路由选项。

步骤 3：更新安全组规则

为了使您的实例能够通过 IPv6 发送和接收流量，您必须更新安全组规则以包含针对 IPv6 地址的规则。例如，在上述示例中，您可以更新您 Web 服务器安全组 (sg-11aa22bb11aa22bb1) 以添加允许来自 IPv6 地址的入站 HTTP、HTTPS 和 SSH 访问的规则。您不需要对数据库安全组的入站规则进行任何更改；允许来自 sg-11aa22bb11aa22bb1 的所有通信的规则包括 IPv6 通信。

要更新入站安全组规则

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导窗格中，选择安全组，并选择您的 Web 服务器安全组。
3. 在入站规则选项卡上，选择编辑入站规则。
4. 对于每条允许 IPv4 流量的规则，请选择添加规则并将该规则配置为允许相应的 IPv6 流量。例如，要添加允许所有通过 IPv6 的 HTTP 流量的规则，对于类型，请选择 HTTP，对于来源，请选择 ::/0。
5. 完成添加标签后，选择保存规则。

更新出站安全组规则

当您将 IPv6 CIDR 块与 VPC 关联时，我们会自动为 VPC 的安全组添加一条允许所有 IPv6 流量的出站规则。但是，如果您修改了安全组的原始出站规则，则不会自动添加此规则，您必须为 IPv6 流量添加等效的出站规则。

更新您的网络 ACL 规则

当您将 IPv6 CIDR 块与 VPC 关联时，我们会自动为默认网络 ACL 添加规则，以允许 IPv6 流量。但是，如果您修改了默认网络 ACL，或者创建了自定义网络 ACL，则必须手动添加 IPv6 流量规则。有关更多信息，请参阅添加和删除规则。

步骤 4：为实例分配 IPv6 地址

当前一代的所有实例类型都支持 IPv6。如果您的实例类型不支持 IPv6，则您必须调整实例的大小以使其成为支持的实例类型，然后再分配 IPv6 地址。您将使用的流程，取决于您所选择的新实例类型是否与当前实例类型兼容。有关更多信息，请参阅《Amazon EC2 用户指南》中的更改实例类型。如果必须从新 AMI 中启动实例来支持 IPv6，可在启动过程中为实例分配 IPv6 地址。

在确认实例类型支持 IPv6 后，可使用 Amazon EC2 控制台为实例分配 IPv6 地址。该 IPv6 地址将分配给实例的主要网络接口（例如，eth0）。有关更多信息，请参阅《Amazon EC2 用户指南》中的为实例分配 IPv6 地址。

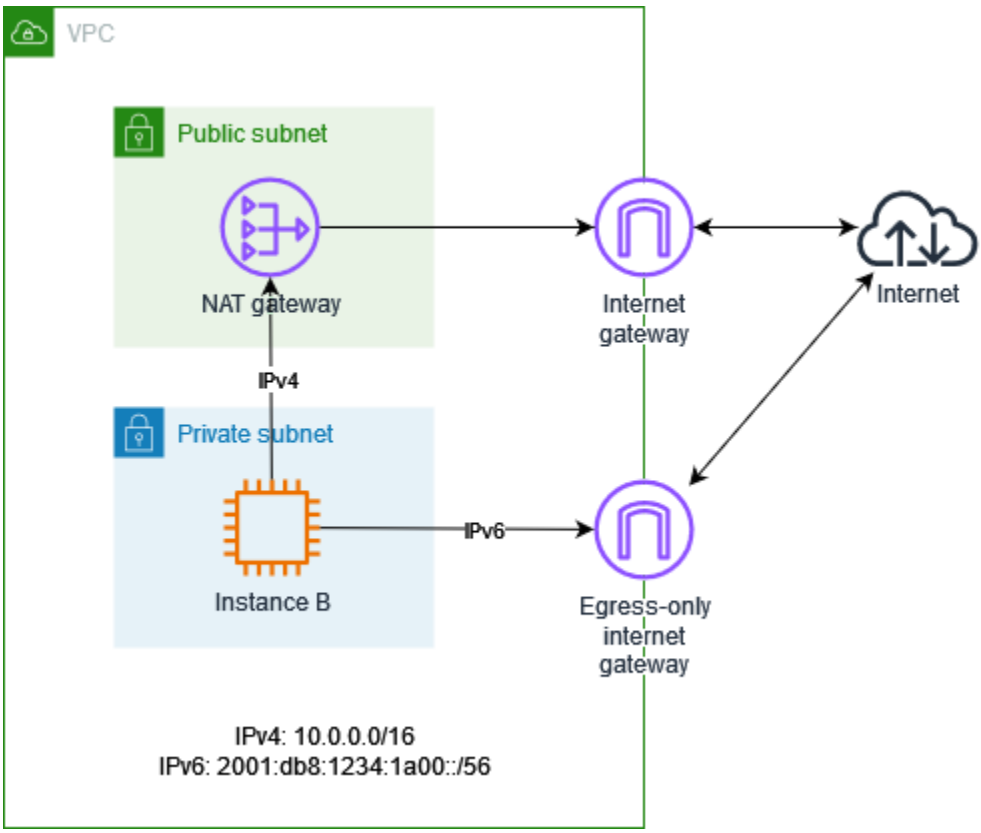
您可以使用 IPv6 地址连接到实例。有关更多信息，请参阅《Amazon EC2 用户指南》中的使用 SSH 客户端连接到 Linux 实例。

如果您使用适用于当前版本操作系统的 AMI 启动了实例，则您的实例已针对 IPv6 进行配置。如果您无法从您的实例执行 IPv6 地址的 ping 操作，则请参阅操作系统的文档来配置 IPv6。

双堆栈 VPC 配置示例

使用双栈配置，您可以同时使用 IPv4 和 IPv6 地址在 VPC 中的资源与互联网上的资源之间进行通信。

下图表示您的 VPC 架构。您的 VPC 具有公有子网和私有子网。VPC 和子网必须同时具有 IPv4 CIDR 块和 IPv6 CIDR 块。私有子网中有一个 EC2 实例，它同时拥有一个 IPv4 地址和一个 IPv6 地址。该实例可以使用 NAT 网关将出站 IPv4 流量发送到互联网，使用仅出口互联网网关将出站 IPv6 流量发送到互联网。



公有子网的路由表

以下是公有子网的路由表。前两个条目是本地路由。第三个条目将所有 IPv4 流量发送到互联网网关。请注意，仅当您计划在公有子网中启动具有 IPv6 地址的 EC2 实例时，才需要第四个条目。

目标位置	目标
VPC IPv4 CIDR	本地
VPC IPv6 CIDR	本地
0.0.0.0/0	*internet-gateway-id*
::/0	*internet-gateway-id*

私有子网的路由表

以下是私有子网的路由表。前两个条目是本地路由。第三个条目将所有 IPv4 流量发送到 NAT 网关。最后一个条目将所有 IPv6 流量发送到仅出口互联网网关。

目标位置	目标
VPC IPv4 CIDR	本地
VPC IPv6 CIDR	本地
0.0.0.0/0	*nat-gateway-id*
::/0	*egress-only-gateway-id*

支持 IPv6 的 AWS 服务

计算机和智能设备使用 IP 地址通过互联网和其他网络相互通信。随着互联网的持续增长，对 IP 地址的需求也在增加。IP 地址最常见的格式是 IPv4。IP 地址的新格式是 IPv6，它可提供比 IPv4 更大的地址空间。

AWS 服务 对 IPv6 的支持包括对双堆栈配置（IPv4 和 IPv6）或仅 IPv6 配置的支持。例如，虚拟私有云（VPC）是您可在其中启动 AWS 资源的 AWS Cloud 的逻辑分隔部分。在 VPC 中，您可以创建仅 IPv4、双堆栈或仅 IPv6 的子网。

AWS 服务 支持通过公共端点进行访问。一些 AWS 服务 还支持使用由 AWS PrivateLink 提供支持的私有端点进行访问。AWS 服务 即使不通过其公共端点支持 IPv6，也可以通过其私有端点支持 IPv6。支持 IPv6 的端点可以使用 AAAA 记录响应 DNS 查询。

支持 IPv6 的服务

下表列出了提供双堆栈支持、仅 IPv6 支持和支持 IPv6 的端点的 AWS 服务。我们将在发布对 IPv6 的其他支持时更新此表。有关某项服务如何支持 IPv6 的详细信息，请参阅适用于该服务的文档。

服务名称	双堆栈支持	仅 IPv6 支持	公共端点支持 IPv6	私有端点支持 IPv6 [1]
AWS Amplify	是	否	是	
Amazon API Gateway	是	否	是	是
AWS App Mesh	是	是	是	否
AWS AppConfig	是	否	是	是

服务名称	双堆栈支持	仅 IPv6 支持	公共端点支持 IPv6	私有端点支持 IPv6 [1]
AWS Application Discovery Service	是	否	是	是
应用程序恢复控制器 (ARC)	是	否	是	
Amazon AppStream 2.0	是	否	否	否
AWS AppSync[2]	部分	否	部分	否
Amazon Athena	是	否	是	是
Amazon Aurora	是	否	是	否
AWS B2B Data Interchange	是	否	是	是
AWS Backup	是	否	是	是
AWS Batch	是	否	是	是
AWS 账单与成本管理 数据导出	是	否	是	是
AWS 账单与成本管理 定价计算器	是	否	是	是
AWS Billing Conductor	是	否	是	是
Amazon Braket	是	是	是	是
AWS Certificate Manager	是	否	是	否
Amazon Comprehend	支持	是	是	是
AWS Clean Rooms	是	是	是	是

服务名称	双堆栈支持	仅 IPv6 支持	公共端点支持 IPv6	私有端点支持 IPv6 [1]
AWS Clean Rooms 机器学习	支持	是	是	是
AWS Cloud9	是	否	是	
AWS 云端控制 API	是	否	是	是
Amazon CloudFront	是	否	否	
AWS CloudHSM	是	否	是	是
AWS CloudTrail	是	否	是	是
Amazon CloudWatch	支持	是	是	是
Amazon CloudWatch Logs	是	是	是	是
Amazon CloudWatch Observability Access Manager	支持	是	是	
Amazon CloudWatch Synthetics	是	否	是	是
AWS Cloud Map	是	是	是	是
AWS Cloud WAN	是	否	是	是
AWS CodeArtifact	是	否	是	是
Amazon Connect Customer Profiles	是	否	是	是
Amazon CodeGuru Profiler	是	否	是	是
AWS Cost Explorer	是	否	是	是

服务名称	双堆栈支持	仅 IPv6 支持	公共端点支持 IPv6	私有端点支持 IPv6 [1]
AWS 成本优化中心	是	否	是	是
AWS Elastic Beanstalk	否	否	是	是
Amazon Cognito	是	否	是	
Amazon Data Firehose	否	否	是	是
Amazon Data Lifecycle Manager	是	否	是	是
AWS Database Migration Service	是	否	否	否
AWS DataSync	是	是	是	是
AWS Deadline Cloud	是	否	是	是
Amazon Detective	支持	是	是	
AWS Direct Connect	是	是	否	是
Amazon EBS 直接 API	是	否	是	是
Amazon EC2	是	是	是	否
Amazon ECR	是	否	是	否
Amazon ECS	是	否	是	是
Amazon EFS	是	是	是	是
Amazon EKS	部分	部分	支持	是
Elastic Load Balancing	部分	部分	否	否
Amazon ElastiCache	是	是	否	否

服务名称	双堆栈支持	仅 IPv6 支持	公共端点支持 IPv6	私有端点支持 IPv6 [1]
AWS 最终用户消息社交	是	否	是	否
AWS Entity Resolution 数据匹配服务	是	否	是	是
AWS Fargate	是	否	否	否
Amazon FSx	否	否	是	是
Amazon GameLift Streams	是	否	是	否
AWS Global Accelerator	是	否	否	
AWS Glue	是	否	否	是
Amazon Managed Grafana [3]	是	否	是	是
AWS Ground Station [4]	是	否	是	是
AWS Identity and Access Management（IAM）	是	是	是	是
AWS IAM Access Analyzer	是	否	是	是
Amazon Inspector	支持	是	是	是
AWS IoT	是	否	是	否
AWS IoT FleetWise	是	否	是	是
AWS IoT SiteWise	是	否	是	是
AWS IoT TwinMaker	是	否	是	是

服务名称	双堆栈支持	仅 IPv6 支持	公共端点支持 IPv6	私有端点支持 IPv6 [1]
AWS IoT Wireless	是	否	是	是
AWS Key Management Service	是	部分	支持	是
Amazon Kinesis Data Streams	是	否	是	否
AWS Lake Formation	否	否	否	是
AWS Lambda	是	否	是	是
Amazon Lightsail	是	是	是	是
Amazon Macie	是	否	是	是
AWS Mainframe Modernization	是	否	是	是
AWS Network Firewall	是	是	否	否
AWS Network Manager	是	否	是	是
Amazon OpenSearch Service	是	否	是	否
AWS Organizations	是	否	是	否
Amazon Personalize	是	否	是	是
Amazon Pinpoint	是	否	是	否
Amazon Polly	是	否	是	是
AWS Private Certificate Authority	是	否	是	是

服务名称	双堆栈支持	仅 IPv6 支持	公共端点支持 IPv6	私有端点支持 IPv6 [1]
AWS Private CA Connector for Active Directory	是	否	是	是
AWS Private CA Connector for SCEP	是	否	是	是
AWS PrivateLink	是	是	是	
Amazon Managed Service for Prometheus	是	否	是	是
Amazon RDS	是	否	是	否
回收站	是	否	是	是
AWS re:Post 私人版	是	否	是	是
AWS Resource Access Manager	是	否	是	是
AWS 资源探索器	是	否	是	是
AWS Resource Groups	是	是	是	是
AWS Resource Groups Tagging API	是	是	是	是
Amazon Route 53	支持	是	否	
Amazon S3	是	否	是	否
AWS Secrets Manager	是	否	是	否
Amazon Security Lake	是	否	是	是
AWS Shield	是	是	否	是

服务名称	双堆栈支持	仅 IPv6 支持	公共端点支持 IPv6	私有端点支持 IPv6 [1]
Amazon Simple Email Service	是	否	是	是
Amazon Simple Notification Service	是	否	是	是
Amazon Simple Queue Service	是	否	是	是
Amazon Simple Workflow Service	是	否	是	是
AWS Site-to-Site VPN	是	否	是	否
AWS Step Functions	是	否	是	是
Amazon Transcribe	支持	是	是	是
AWS Transfer Family [5]	是	否	是	是
AWS Transit Gateway	是	否	是	否
Amazon Translate	支持	是	是	是
Amazon Verified Permissions	是	否	是	是
Amazon VPC	是	是	是	否
AWS WAF	是	是	否	是
Amazon WorkSpaces	是	否	否	否
AWS X-Ray	是	否	是	是
EC2 Image Builder	支持	是	是	是

[1] 空单元格表示该服务未与 AWS PrivateLink 集成。

[2] 此条目表示通过 AWS AppSync SDK API 对 AWS AppSync GraphLQL 和事件 API 配置操作的 IPv6 支持。与客户托管的 AWS AppSync GraphQL 和事件 API 的客户端连接不支持 IPv6。

[3] 此条目表示对 Grafana 工作区管理操作（例如更新工作区和工作区权限）的 IPv6 支持。常规 Grafana 工作区操作（例如创建和编辑仪表板或查询数据源）不支持 IPv6。

[4] 此条目表示对 AWS Ground Station 控制面板操作（例如调用 AWS Ground Station API）的 IPv6 支持。AWS Ground Station 数据面板不支持 IPv6，因此请确保您要向其传输数据的资源（例如 Amazon EC2 实例）可以通过 IPv4 访问。

[5] 有关 AWS Transfer Family 中 IPv6 支持的更多详细信息，请参阅 IPv6 限制。

其他 IPv6 支持

计算

- Amazon EC2 支持在仅限 IPv6 的子网中启动基于 Nitro 系统的实例。
- Amazon EC2 可为实例元数据服务（IMDS）和 Amazon Time Sync Service 提供 IPv6 端点。

联网和内容分发

- Amazon VPC 支持创建仅限 IPv6 的子网。
- 通过支持子网上的 DNS64 和 NAT 网关上的 NAT64，Amazon VPC 可有助于 IPv6 AWS 资源与 IPv4 资源通信。

安全、身份和合规性

- 在与网络相关的调查发现和实体配置文件中，Amazon Detective 支持 IPv6 地址。
- AWS Identity and Access Management（IAM）在基于 IAM 身份的策略中支持 IPv6 地址。
- Amazon Macie 支持个人身份信息（PII）中的 IPv6 地址。
- Amazon Security Lake 支持在日志源和订阅用户的所有操作中使用 IPv6 地址。

管理和治理

- AWS CloudTrail 记录包括源 IPv6 信息。
- AWS CLI v2 支持仅限 IPv6 的客户端通过 IPv6 连接进行下载。

了解更多

- AWS 上的 IPv6
- 双堆栈和仅 IPv6 Amazon VPC 参考架构（PDF）

配置虚拟私有云

Amazon Virtual Private Cloud（VPC）属于基本构建基块，可帮助您在 AWS 云中预置逻辑隔离的虚拟网络。您可以通过创建自己的 VPC 来完全控制联网环境，包括定义 IP 地址范围、子网、路由表和连接选项的能力。

您的 AWS 账户包含每个 AWS 区域的默认 VPC。此默认 VPC 预配置了相应设置，使其成为快速启动资源的便捷选项。不过，默认 VPC 不一定总能符合您的长期联网需求。这就是创建额外 VPC 的优势所在。

与依赖于随每个新 AWS 账户预置的默认 VPC 相比，创建额外 VPC 具有多种优势。借助自行管理的 VPC，无论是实施多层应用程序、连接到本地资源，还是按部门或业务单位隔离工作负载，您都可以架构网络拓扑，使之与自身特定要求严格一致。

此外，创建多个 VPC 可以提高不同应用程序或业务单位之间的安全性和隔离性。每个 VPC 都充当独立的虚拟网络，使您能够应用针对每个环境量身定制的不同安全策略、访问控制和路由配置。

归根结底，使用默认 VPC 或创建一个（或多个）自定义 VPC 的决策应基于自身特定的应用程序要求、安全需求和长期可扩展性目标。投入时间精心设计 VPC 基础设施，可以带来强大、安全和适应性强的云联网基础方面的优势。

内容

- VPC 基础知识
- VPC 配置选项
- 默认 VPC
- 创建 VPC
- 可视化 VPC 中的资源
- 将 CIDR 块添加到 VPC 或从中删除
- Amazon VPC 中的 DHCP 选项集
- VPC 中的 DNS 属性
- VPC 的网络地址用量
- 与其他账户共享 VPC 子网
- 将 VPC 扩展到本地区域、Wavelength 区域或 Outpost
- 删除您的 VPC
- 使用 Console-to-Code，通过您的 VPC 控制台操作生成基础设施即代码

VPC 基础知识

VPC 涵盖一个区域中的所有可用区。在创建 VPC 之后，您可以在每个可用区域中添加一个或多个子网。有关更多信息，请参阅 子网。

内容

- VPC IP 地址范围
- VPC 图
- VPC 资源

VPC IP 地址范围

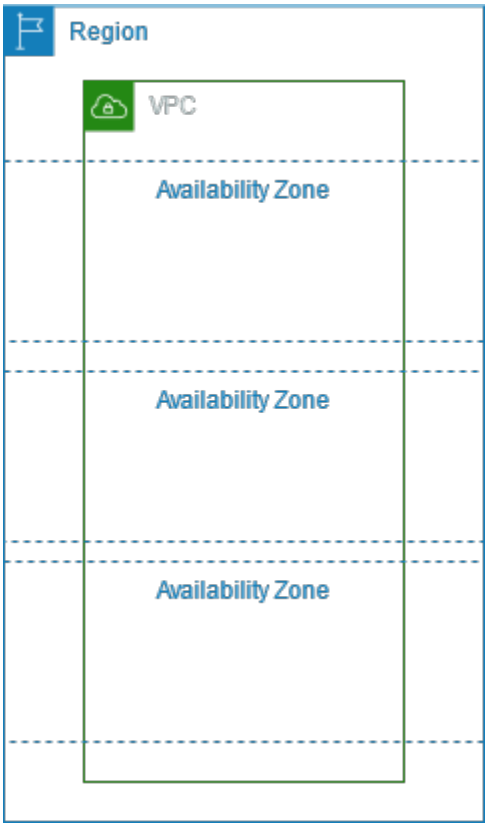
创建 VPC 时，您可以按以下方式指定其 IP 地址：

- 仅 IPv4 – VPC 具有 IPv4 CIDR 块，但没有 IPv6 CIDR 块。
- 双堆栈 – VPC 同时具有 IPv4 CIDR 块和 IPv6 CIDR 块。

有关更多信息，请参阅 为 VPC 和子网分配 IP 地址。

VPC 图

下图显示了一个没有额外 VPC 资源的 VPC。有关 VPC 示例配置，请参阅 示例。



VPC 资源

每个 VPC 会自动提供以下资源：

- 默认 DHCP 选项集
- 默认网络 ACL
- 默认安全组
- 主路由表

您可以为您的 VPC 创建下列资源：

- 网络 ACL
- 自定义路由表
- 安全组
- Internet 网关
- NAT 网关

VPC 配置选项

您可以在创建 VPC 时指定以下配置选项。

可用区

在一个 AWS 区域中具有冗余电源、联网和连接的分散数据中心。通过使用多个可用区，您可以获得比单个数据中心具有更高可用性、容错能力和可扩展性的生产级应用程序和数据库。通过将应用程序进行分区，在跨可用区的子网中运行，可以实现更好的隔离和保护，防止停电、雷击、龙卷风、地震等问题的影响。

CIDR 块

您必须为您的 VPC 和子网指定 IP 地址范围。有关更多信息，请参阅 为 VPC 和子网分配 IP 地址。

DNS 选项

如果您需要在子网中启动的 EC2 实例使用公有 IPv4 DNS 主机名，则必须同时启用这两个 DNS 选项。有关更多信息，请参阅 VPC 中的 DNS 属性。

- 启用 DNS 主机名：在 VPC 中启动的 VPC 实例将接收与其公有 IPv4 地址对应的公有 DNS 主机名。
- 启用 DNS 解析：私有 DNS 主机名的 DNS 解析由名为 Route 53 Resolver 的 Amazon DNS 服务器为 VPC 提供。

互联网网关

将您的 VPC 连接到互联网。公有子网中的实例可以访问互联网，因为子网路由表包含一条将指向互联网的流量发送到互联网网关的路由。如果服务器不需要直接从互联网访问，则不应将其部署到公有子网中。详情请见 互联网网关。

名称

您为 VPC 和其他 VPC 资源指定的名称将用于创建名称标签。如果您使用控制台中的名称标签自动生成功能，则标签值的格式为 *name-resource*。

NAT 网关

让私有子网中的实例能够将出站流量发送到互联网，但阻止互联网上的资源连接到实例。在生产环境中，我们建议您在每个活动可用区中部署一个 NAT 网关。有关更多信息，请参阅 NAT 网关。

路由表

包含一组被称为路由的规则，用于决定来自您的子网或网关的网络流量将指向何处。有关更多信息，请参阅 路由表。

子网

您的 VPC 内的一个 IP 地址范围。您可以在子网中启动 AWS 资源（如 EC2 实例）。每个子网都完全位于一个可用区之内。通过在至少两个可用区内启动实例，应用程序将不受单一可用区故障的影响。

公有子网有一条指向互联网网关的直接路由。公有子网中的资源可以访问公有互联网。私有子网不具有指向互联网网关的直接路由。私有子网中的资源需要使用另一个组件（例如 NAT 设备）才能访问公共互联网。

有关更多信息，请参阅 子网。

租赁

此选项定义您启动到此 VPC 中的 EC2 实例是在与其他 AWS 账户 共享的硬件上运行，还是在专供您使用的硬件上运行。如果您选择 VPC 的租赁为 `Default`，则启动到此 VPC 的 EC2 实例将使用您在启动实例时指定的租赁属性 – 有关更多信息，请参阅《Amazon EC2 用户指南》中的使用定义的参数启动实例。如果您选择 VPC 的租赁为 `Dedicated`，则这些实例将始终在专供您使用的硬件上作为专用实例运行。如果您使用的是 AWS Outpost，则您的 Outpost 需要私有连接；您必须使用 `Default` 租赁。

默认 VPC

当您开启使用 Amazon VPC 时，每个 AWS 区域都有一个原定设置的 VPC。原定设置 VPC 在每个可用区中都有一个公有子网、一个互联网网关以及用于启用 DNS 解析的设置。因此，您可以立即在原定设置 VPC 中启动 Amazon EC2 实例。您还可以在默认 VPC 中使用 Elastic Load Balancing、Amazon RDS 和 Amazon EMR 等服务。

原定设置 VPC 适用于快速入门和启动公有实例 (如博客或简单的网站)。您可以按需修改您的默认 VPC 的组件。

您可以将子网添加到原定设置 VPC 中。有关更多信息，请参阅 the section called “创建子网”。

内容

- 默认 VPC 组件
- 默认子网
- 查看默认 VPC 和默认子网

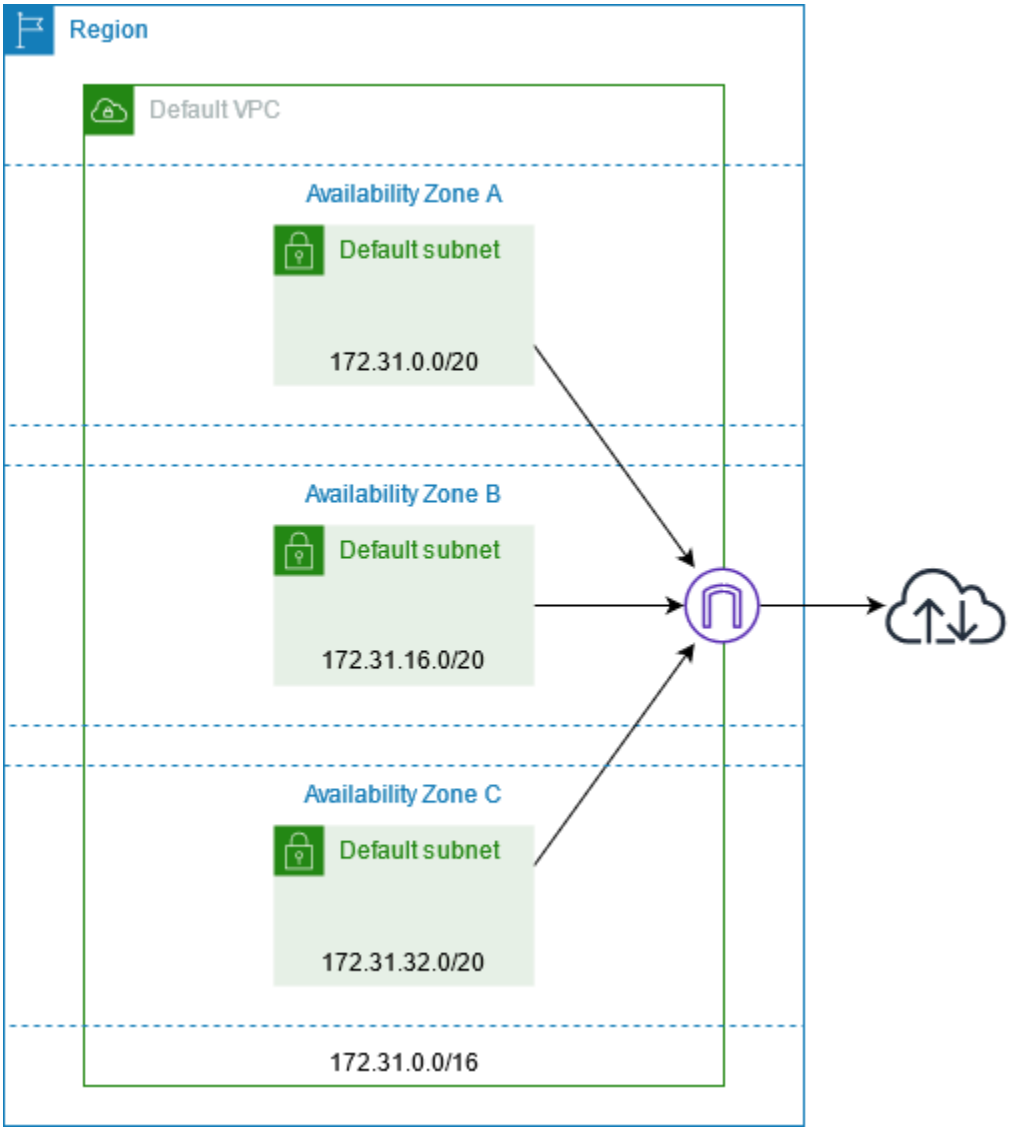
默认 VPC 组件

当我们创建默认 VPC 时，我们会通过以下操作为您完成设置：

- 创建 IPv4 CIDR 块大小为 /16 的 VPC (172.31.0.0/16)。最多可提供 65536 个私有 IPv4 地址。
- 在每个可用区内创建大小为 /20 的默认子网。这将为每个子网提供多达 4,096 个地址，其中有一些被预留下来供我们使用。
- 创建互联网网关并将其连接到您的默认 VPC。
- 在主路由表中添加一个将所有流量 (0.0.0.0/0) 指向互联网网关的路由。
- 创建默认安全组并将其与您的默认 VPC 关联。
- 创建默认网络访问控制列表 (ACL)，并将其与您的默认 VPC 关联。
- 关联为您的 AWS 账户设置的默认 DHCP 选项与您的默认 VPC。

> **Note**
>
> - Amazon 代表您创建上述资源。IAM 策略不应用到这些操作，因为您不执行这些操作。例如，如果您有 IAM 策略拒绝了调用 CreateInternetGateway 的功能，然后您调用 CreateDefaultVpc，则仍会在默认 VPC 中创建互联网网关。为了防止 Amazon 创建互联网网关，您必须拒绝 CreateDefaultVpc 和 CreateInternetGateway。
> - 要阻止您账户中往返于互联网网关的所有流量，请参阅屏蔽 VPC 和子网的公共访问权限。

下图表明了我们为默认 VPC 设置的关键组件。



下表显示默认 VPC 的主路由表中的路由。

目标位置	目标
172.31.0.0/16	本地
0.0.0.0/0	*internet_gateway_id*

您可以像使用任何其他 VPC 一样使用默认 VPC：

- 添加更多非默认子网。
- 修改主路由表。

- 添加更多路由表。
- 关联更多安全组。
- 更新默认安全组的规则。
- 添加 AWS Site-to-Site VPN 连接。
- 添加更多 IPv4 CIDR 块。
- 使用 Direct Connect 网关访问远程区域中的 VPC。有关 Direct Connect 网关选项的信息，请参阅《AWS Direct Connect 用户指南》中的 Direct Connect 网关。

您可像使用任何其他子网一样使用默认子网；可添加自定义路由表和设置网络 ACL。您还可以在启动 EC2 实例时指定特定默认子网。

您可以选择将 IPv6 CIDR 块与默认 VPC 关联。

默认子网

默认情况下，默认子网为公有子网，因为主路由表会将指定发往 Internet 的子网流量发送到互联网网关。您可以从到互联网网关的目标 0.0.0.0/0 中删除路由，以使默认子网变为私有子网。但是，如果您执行此操作，则在该子网中运行的所有 EC2 实例都无法访问 Internet。

您在默认子网中启动的实例将同时接收公有 IPv4 地址和私有 IPv4 地址以及公有和私有 DNS 主机名。在默认 VPC 中的非默认子网内启动的实例不接收公有 IPv4 地址或 DNS 主机名。您可以更改您子网的默认公有 IP 寻址行为。有关更多信息，请参阅 修改子网的 IP 寻址属性。

有时，AWS 可能会向某个区域添加新可用区。大多数情况下，我们会在几天内在此可用区中为您的默认 VPC 自动创建新的默认子网。但是，如果您对默认 VPC 进行过任何修改，那么我们不会添加新的默认子网。如果您希望对新的可用区使用默认子网，则可以自行创建一个。有关更多信息，请参阅 创建默认子网。

查看默认 VPC 和默认子网

本节旨在介绍如何使用默认 VPC 和默认子网。

内容

- 查看您的默认 VPC 和默认子网
- 创建默认 VPC
- 创建默认子网
- 删除您的默认子网和默认 VPC

查看您的默认 VPC 和默认子网

您可以使用 Amazon VPC 控制台或命令行查看您的默认 VPC 和子网。

使用控制台查看您的默认 VPC 和子网

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Your VPCs。
3. 在 Default VPC 列中，查找值 Yes。记下默认 VPC 的 ID。
4. 在导航窗格中，选择 Subnets (子网)。
5. 在搜索栏中，键入默认 VPC 的 ID。返回的子网是您的默认 VPC 中的子网。
6. 要验证哪些子网是默认子网，请在 Default Subnet 列中查找值 Yes。

使用命令行描述您的默认 VPC

- 使用 describe-vpcs (AWS CLI)
- 使用 Get-EC2Vpc (AWS Tools for Windows PowerShell)

将命令与 `isDefault` 筛选器结合使用并将筛选值设置为 `true`。

使用命令行描述您的默认子网

- 使用 describe-subnets (AWS CLI)
- 使用 Get-EC2Subnet (AWS Tools for Windows PowerShell)

将命令与 `vpc-id` 筛选器结合使用并将筛选值设置为默认 VPC 的 ID。在输出中，`DefaultForAz` 字段对默认子网设置为 `true`。

创建默认 VPC

如果您删除了默认 VPC，则可以创建一个新的默认 VPC。您无法恢复之前删除的默认 VPC，并且无法将现有非默认 VPC 标记为默认 VPC。

当您创建默认 VPC 时，将使用默认 VPC 的标准组件创建它 (包括每个可用区中的默认子网)。您无法指定您自己的组件。新的默认 VPC 的子网 CIDR 块可能不会与之前的默认 VPC 映射到同一可用区。例如，如果具有 CIDR 块的子网 `172.31.0.0/20` 是在之前的默认 VPC 的 `us-east-2a` 中创建的，则该子网可能在新的默认 VPC 的 `us-east-2b` 中创建。

如果您在该区域中已经有默认 VPC，则无法创建另一个默认 VPC。

使用控制台创建默认 VPC

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Your VPCs。
3. 依次选择 Actions 和 Create Default VPC。
4. 选择创建。关闭确认屏幕。

使用命令行创建默认 VPC

您可以使用 create-default-vpc AWS CLI 命令。此命令没有任何输入参数。

```
aws ec2 create-default-vpc
```

下面是示例输出。

```
{
    "Vpc": {
        "VpcId": "vpc-3f139646",
        "InstanceTenancy": "default",
        "Tags": [],
        "Ipv6CidrBlockAssociationSet": [],
        "State": "pending",
        "DhcpOptionsId": "dopt-61079b07",
        "CidrBlock": "172.31.0.0/16",
        "IsDefault": true
    }
}
```

或者，您可以将 New-EC2DefaultVpc 工具用于 Windows PowerShell 命令或 CreateDefaultVpc Amazon EC2 API 操作。

创建默认子网

> **Note**
>
> 您不能使用 AWS Management Console 创建默认子网。

您可以在没有默认子网的可用区中创建一个默认子网。例如，如果您已删除默认子网，或者 AWS 已添加新的可用区但未在您的默认 VPC 中为该区域自动创建默认子网，则您可能需要创建一个默认子网。

创建默认子网时，将使用您的默认 VPC 中的下一个可用连续空间内的大小为 /20 的 IPv4 CIDR 块创建它。以下规则适用：

- 不能自行指定 CIDR 块。
- 不能恢复已删除的之前的默认子网。
- 每个可用区只能有一个默认子网。
- 不能在非默认 VPC 中创建默认子网。

如果您的默认 VPC 中没有用于创建大小为 /20 的 CIDR 块的足够地址空间，则请求会失败。如果您需要更多地址空间，则可以将 IPv4 CIDR 块添加到您的 VPC。

如果您已将 IPv6 CIDR 块与您的默认 VPC 关联，则新的默认子网不会自动接收 IPv6 CIDR 块。您可以改为在创建一个 IPv6 CIDR 块后将其与默认子网关联。有关更多信息，请参阅 将 IPv6 CIDR 块添加到子网或从中删除。

使用 AWS CLI 创建默认子网

使用 create-default-subnet AWS CLI 命令并指定要在其中创建子网的可用区。

```
aws ec2 create-default-subnet --availability-zone us-east-2a
```

下面是示例输出。

```
{
    "Subnet": {
        "AvailabilityZone": "us-east-2a",
        "Tags": [],
        "AvailableIpAddressCount": 4091,
        "DefaultForAz": true,
        "Ipv6CidrBlockAssociationSet": [],
        "VpcId": "vpc-1a2b3c4d",
        "State": "available",
        "MapPublicIpOnLaunch": true,
        "SubnetId": "subnet-1122aabb",
        "CidrBlock": "172.31.32.0/20",
        "AssignIpv6AddressOnCreation": false
    }
```

```
}
```

有关设置 AWS CLI 的更多信息，请参阅《AWS Command Line Interface 用户指南》。

或者，您也可以使用 New-EC2DefaultSubnet Tools for Windows PowerShell 命令或 CreateDefaultSubnet Amazon EC2 API 操作。

删除您的默认子网和默认 VPC

您可以像删除其他任何子网或 VPC 一样删除默认子网或默认 VPC。但是，如果您删除默认子网或默认 VPC，则在启动实例时必须显式指定您的一个 VPC 中的一个子网。如果您没有另一个 VPC，则必须创建至少在一个可用区内有一个子网的 VPC。有关更多信息，请参阅 创建 VPC。

如果您删除了默认 VPC，则可以创建一个新的默认 VPC。有关更多信息，请参阅 创建默认 VPC。

如果您删除默认子网，则可以创建一个新的默认子网。有关更多信息，请参阅 创建默认子网。为了确保您的新默认子网按预期正常运行，请修改子网属性以将公有 IP 地址分配到在该子网中启动的实例。有关更多信息，请参阅 修改子网的 IP 寻址属性。每个可用区只能有一个默认子网。不能在非默认 VPC 中创建默认子网。

创建 VPC

按照以下过程创建虚拟私有云（VPC）。VPC 必须有额外的资源，例如子网、路由表和网关，然后才能在 VPC 中创建 AWS 资源。

内容

- 创建 VPC 以及其他 VPC 资源
- 仅创建 VPC
- 使用 AWS CLI 创建 VPC

有关修改 VPC 的信息，请参阅the section called “添加或删除 CIDR 块”。

创建 VPC 以及其他 VPC 资源

按照以下过程创建 VPC 以及运行应用程序所需的其他 VPC 资源，例如子网、路由表、互联网网关和 NAT 网关。有关 VPC 示例配置，请参阅 示例。

使用控制台创建 VPC、子网和其他 VPC 资源

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。

2. 在 VPC 控制面板上，选择创建 VPC。
3. 对于要创建的资源，选择 VPC 等。
4. 保持选中名称标签自动生成以为 VPC 资源创建名称标签，或者清除此选项以为 VPC 资源提供您自己的名称标签。
5. 对于 IPv4 CIDR 块，输入 VPC 的 IPv4 地址范围。VPC 必须具有一个 IPv4 地址范围。
6. （可选）要支持 IPv6 流量，请选择 IPv6 CIDR 块，然后选择 Amazon 提供的 IPv6 CIDR 块。
7. 选择租赁选项。此选项定义您启动到此 VPC 中的 EC2 实例是在与其他 AWS 账户 共享的硬件上运行，还是在专供您使用的硬件上运行。如果您选择将 VPC 的租赁设为 `Default`，则在此 VPC 中启动的 EC2 实例将使用您在启动实例时指定的租赁属性。有关更多信息，请参阅《Amazon EC2 用户指南》中的使用定义的参数启动实例。如果您选择 VPC 的租赁为 `Dedicated`，则这些实例将始终在专供您使用的硬件上作为专用实例运行。如果您使用的是 AWS Outpost，则您的 Outpost 需要私有连接；您必须使用 `Default` 租赁。
8. 对于可用区（AZ）数量，我们建议生产环境至少在两个可用区域中预置子网。要为您的子网选择可用区，请展开自定义可用区。否则可让 AWS 为您选择。
9. 要配置子网，请选择公有子网的数量和私有子网的数量的值。要选择子网的 IP 地址范围，请展开自定义子网 CIDR 块。否则可让 AWS 为您选择。
10. （可选）如果私有子网中的资源需要通过 IPv4 访问公共互联网，则对于 NAT 网关，请选择要在其中创建 NAT 网关的可用区数量。在生产环境中，我们建议您在每个可用区部署一个 NAT 网关，其中包含需要访问公共互联网的资源。请注意，使用 NAT 网关会产生成本。有关更多信息，请参阅 适用于 NAT 网关的定价。
11. （可选）如果私有子网中的资源需要通过 IPv6 访问公共互联网，对于仅限出口的互联网网关，请选择是。
12. （可选）如果您需要直接从 VPC 访问 Amazon S3，请选择 VPC 端点、S3 网关。这将为 Amazon S3 创建一个网关 VPC 端点。有关更多信息，请参阅《AWS PrivateLink 指南》中的 Gateway endpoints。
13. （可选）对于 DNS 选项，默认情况下，两个域名解析选项均处于启用状态。如果默认设置无法满足您的需求，您可以禁用这些选项。
14. （可选）要向 VPC 添加标签，请展开其他标签，选择添加新标签，然后输入标签键和标签值。
15. 在预览窗格中，您可以直观地显示您所配置的 VPC 资源之间的关系。实线表示资源之间的关系。虚线表示指向 NAT 网关、互联网网关和网关端点的网络流量。创建 VPC 后，您可以使用资源地图选项卡，随时以此格式直观地显示 VPC 的资源。有关更多信息，请参阅 可视化 VPC 中的资源。
16. 配置完 VPC 后，选择创建 VPC。

仅创建 VPC

按照以下过程，使用 Amazon VPC 控制台创建无额外 VPC 资源的 VPC。

使用控制台创建无额外 VPC 资源的 VPC。

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在 VPC 控制面板上，选择创建 VPC。
3. 对于要创建的资源，选择 仅 VPC。
4. （可选）对于名称标签，输入 VPC 的名称。这样做可创建具有 Name 键以及您指定的值的标签。
5. 对于 IPv4 CIDR block（IPv4 CIDR 块），请执行以下操作之一：
   - 选择 IPv4 CIDR 手动输入，然后输入您的 VPC 的 IPv4 地址范围。
   - 选择 IPAM 分配的 IPv4 CIDR 块，然后选择您的 Amazon VPC IP 地址管理器（IPAM）IPv4 地址池和网络掩码。CIDR 块的大小受 IPAM 池上的分配规则限制。IPAM 是一项 VPC 功能，可让您更轻松地计划、跟踪和监控 AWS 工作负载的 IP 地址。有关更多信息，请参阅《Amazon VPC IPAM 用户指南》https://docs.aws.amazon.com/vpc/latest/ipam/what-it-is-ipam.html。

     如果您使用 IPAM 来管理 IP 地址，我们建议您选择此选项。否则，您为 VPC 指定的 CIDR 块可能与 IPAM CIDR 分配重叠。
6. （可选）要创建双堆栈 VPC，请为您的 VPC 指定一个 IPv6 地址范围。对于 IPv6 CIDR block（IPv6 CIDR 块），请执行以下操作之一：

   - 如果使用 Amazon VPC IP 地址管理器，并且需要从 IPAM 池预置 IPv6 CIDR，则选择 IPAM 分配的 IPv6 CIDR 块。如果您使用 IPAM 分配的 IPv6 CIDR 块为 VPC 预置 IPv6 CIDR，则可以利用连续的 IPv6 CIDR 来创建 VPC。连续分配的 CIDR 是按顺序分配的 CIDR。它们使您能够简化安全和网络规则；IPv6 CIDR 可以跨网络和安全结构（如访问控制列表、路由表、安全组和防火墙）聚合在单个条目中。

     您可以通过两个选项，在 CIDR 块下为 VPC 预置一个 IP 地址范围：

     - 网络掩码长度：选择此选项可为 CIDR 选择网络掩码长度。请执行以下操作之一：
       - 如果已为 IPAM 池选择默认网络掩码长度，则可以选择默认为 IPAM 网络掩码长度，以使用 IPAM 管理员为 IPAM 池设置的默认网络掩码长度。有关可选默认网络掩码长度分配规则的更多信息，请参阅《Amazon VPC IPAM 用户指南》中的创建区域 IPv6 池。

- 如果未为 IPAM 池选择默认网络掩码长度，则选择一个比 IPAM 池 CIDR 的网络掩码长度更具体的网络掩码长度。例如，假设 IPAM 池 CIDR 为 /50，则可以为 VPC 选择介于 /52 至 /60 之间的网络掩码长度。可能的网络掩码长度介于 /44 和 /60 之间，增量为 /4。
- 选择 CIDR：选择此选项可手动输入 IPv6 地址。您只能选择比 IPAM 池 CIDR 的网络掩码长度更具体的网络掩码长度。例如，假设 IPAM 池 CIDR 为 /50，则可以为 VPC 选择介于 /52 至 /60 之间的网络掩码长度。可能的 IPv6 网络掩码长度介于 /44 和 /60 之间，增量为 /4。

- 选择 Amazon 提供的 IPv6 CIDR 块，以从 Amazon 的 IPv6 地址池请求 IPv6 CIDR 块。对于 Network Border Group（网络边界组），选择 AWS 从中通告 IP 地址的组。Amazon 提供 /56 固定大小的 IPv6 CIDR 块。
- 选择我拥有的 IPv6 CIDR，以预置您已经带到 AWS 的 IPv6 CIDR。有关自带 IP 地址范围到 AWS 的更多信息，请参阅《Amazon EC2 用户指南》中的自带 IP 地址（BYOIP）。您可以使用以下 CIDR 块选项为 VPC 预置 IP 地址范围：
  - 无偏好：选择此选项使用 /56 的网络掩码长度。
  - 选择 CIDR：选择此选项可手动输入 IPv6 地址，然后选择比 BYOIP CIDR 的大小更具体的网络掩码长度。例如，假设 BYOIP 池 CIDR 为 /50，则可以为 VPC 选择介于 /52 至 /60 之间的网络掩码长度。可能的 IPv6 网络掩码长度介于 /44 和 /60 之间，增量为 /4。

7. （可选）选择租赁选项。此选项定义您启动到此 VPC 中的 EC2 实例是在与其他 AWS 账户 共享的硬件上运行，还是在专供您使用的硬件上运行。如果您选择 VPC 的租赁为 `Default`，则启动到此 VPC 的 EC2 实例将使用您在启动实例时指定的租赁属性 – 有关更多信息，请参阅《Amazon EC2 用户指南》中的使用定义的参数启动实例。如果您选择 VPC 的租赁为 `Dedicated`，则这些实例将始终在专供您使用的硬件上作为专用实例运行。如果您使用的是 AWS Outpost，则您的 Outpost 需要私有连接；您必须使用 `Default` 租赁。
8. （可选）要向 VPC 添加标签，请选择添加新标签，然后输入标签键和标签值。
9. 选择创建 VPC。
10. 创建 VPC 后，您可以添加子网。有关更多信息，请参阅 创建子网。

使用 AWS CLI 创建 VPC

以下过程包含创建 VPC 的示例 AWS CLI 命令以及运行应用程序所需的其他 VPC 资源。如果您运行此过程中的所有命令，您将创建一个 VPC、一个公有子网、一个私有子网、针对每个子网的路由表、一个互联网网关、一个仅限出口的互联网网关和一个公有 NAT 网关。如果您不需要所有这些资源，则可以仅使用您需要的示例命令。

先决条件

在开始之前，请安装并配置 AWS CLI。配置 AWS CLI 时，系统会提示您输入 AWS 凭证。本过程中的示例假定您已配置好默认区域。否则，请为每个命令添加 `--region` 选项。有关更多信息，请参阅安装或更新 AWS CLI和配置 AWS CLI。

标记

在创建资源后，您可以使用 create-tags 命令为资源添加标签。您还可以按如下方式将 `--tag-specification` 选项添加到资源的创建命令中。

```
--tag-specifications ResourceType=vpc,Tags=[{Key=Name,Value=my-project}]
```

使用 AWS CLI 创建 VPC 以及其他 VPC 资源

1. 使用下面的 create-vpc 命令创建具有指定 IPv4 CIDR 块的 VPC。

   ```
   aws ec2 create-vpc --cidr-block 10.0.0.0/24 --query Vpc.VpcId --output text
   ```

   要创建双堆栈 VPC，请如以下示例所示，添加 `--amazon-provided-ipv6-cidr-block` 选项以添加 Amazon 提供的 IPv6 CIDR 块。

   ```
   aws ec2 create-vpc --cidr-block 10.0.0.0/24 --amazon-provided-ipv6-cidr-block --
   query Vpc.VpcId --output text
   ```

   这些命令将返回新 VPC 的 ID。示例如下：

   ```
   vpc-1a2b3c4d5e6f1a2b3
   ```

2. [双堆栈 VPC] 使用以下 describe-vpcs 命令获取与 VPC 关联的 IPv6 CIDR 块。

   ```
   aws ec2 describe-vpcs --vpc-id vpc-1a2b3c4d5e6f1a2b3 --query
    Vpcs[].Ipv6CidrBlockAssociationSet[].Ipv6CidrBlock --output text
   ```

   下面是示例输出。

   ```
   2600:1f13:cfe:3600::/56
   ```

3. 根据您的使用场景创建一个或多个子网。在生产环境中，我们建议您至少在两个可用区中启动资源。使用下面的任意一种命令创建每个子网。

   - 仅 IPv4 子网 – 要创建具有特定 IPv4 CIDR 块的子网，请使用下面的 create-subnet 命令。

```
aws ec2 create-subnet --vpc-id vpc-1a2b3c4d5e6f1a2b3 --cidr-block 10.0.1.0/20
 --availability-zone us-east-2a --query Subnet.SubnetId --output text
```

- 双堆栈子网 – 如果您创建了双堆栈 VPC，则可以如以下命令所示，使用 `--ipv6-cidr-block` 选项创建双堆栈子网。

```
aws ec2 create-subnet --vpc-id vpc-1a2b3c4d5e6f1a2b3 --cidr-block 10.0.1.0/20
 --ipv6-cidr-block 2600:1f13:cfe:3600::/64 --availability-zone us-east-2a --
query Subnet.SubnetId --output text
```

- 仅 IPv6 子网 – 如果您创建了双堆栈 VPC，则可以如以下命令所示，使用 `--ipv6-native` 选项创建仅 IPv6 子网。

```
aws ec2 create-subnet --vpc-id vpc-1a2b3c4d5e6f1a2b3 --ipv6-native --ipv6-
cidr-block 2600:1f13:cfe:3600::/64 --availability-zone us-east-2a --query
 Subnet.SubnetId --output text
```

这些命令将返回新子网的 ID。示例如下：

```
subnet-1a2b3c4d5e6f1a2b3
```

4. 如果您的 Web 服务器或 NAT 网关需要公有子网，请执行以下操作：

   a. 使用以下 create-internet-gateway 命令创建互联网网关。该命令将返回新互联网网关的 ID。

```
aws ec2 create-internet-gateway --query InternetGateway.InternetGatewayId --
output text
```

   b. 使用以下 attach-internet-gateway 命令将互联网网关附加到 VPC。使用上一步返回的互联网网关 ID。

```
aws ec2 attach-internet-gateway --vpc-id vpc-1a2b3c4d5e6f1a2b3 --internet-
gateway-id igw-id
```

   c. 使用以下 create-route-table 命令为公有子网创建自定义路由表。该命令将返回新路由表的 ID。

```
aws ec2 create-route-table --vpc-id vpc-1a2b3c4d5e6f1a2b3 --query
 RouteTable.RouteTableId --output text
```

d. 使用以下 create-route 命令，在路由表中创建一条会将所有 IPv4 流量发送到互联网网关的路由。使用公有子网的路由表 ID。

```
aws ec2 create-route --route-table-id rtb-id-public --destination-cidr-block
 0.0.0.0/0 --gateway-id igw-id
```

e. 使用以下 associate-route-table 命令将路由表关联到公有子网。使用公有子网的路由表 ID 和公有子网的 ID。

```
aws ec2 associate-route-table --route-table-id rtb-id-public --subnet-
id subnet-id-public-subnet
```

5. [IPv6] 您可以添加仅限出口的互联网网关，以确保私有子网中的实例可以通过 IPv6 访问互联网（例如，获取软件更新），但互联网上的主机无法访问您的实例。

a. 使用以下 create-egress-only-internet-gateway 命令创建仅限出口的互联网网关。该命令将返回新互联网网关的 ID。

```
aws ec2 create-egress-only-internet-gateway --vpc-id vpc-1a2b3c4d5e6f1a2b3 --
query EgressOnlyInternetGateway.EgressOnlyInternetGatewayId --output text
```

b. 使用以下 create-route-table 命令为私有子网创建自定义路由表。该命令将返回新路由表的 ID。

```
aws ec2 create-route-table --vpc-id vpc-1a2b3c4d5e6f1a2b3 --query
 RouteTable.RouteTableId --output text
```

c. 使用以下 create-route 命令，在私有子网的路由表中创建一条会将所有 IPv6 流量发送到仅限出口的互联网网关的路由。使用上一步中返回的路由表 ID。

```
aws ec2 create-route --route-table-id rtb-id-private --destination-cidr-
block ::/0 --egress-only-internet-gateway eigw-id
```

d. 使用以下 associate-route-table 命令将路由表关联到私有子网。

```
aws ec2 associate-route-table --route-table-id rtb-id-private --subnet-
id subnet-id-private-subnet
```

6. 如果您需要为私有子网中的资源使用 NAT 网关，请执行以下操作：

a. 使用以下 allocate-address 命令为 NAT 网关创建弹性 IP 地址。

```
aws ec2 allocate-address --domain vpc --query AllocationId --output text
```

b. 使用以下 create-nat-gateway 命令在公有子网中创建 NAT 网关。使用上一步返回的分配 ID。

```
aws ec2 create-nat-gateway --subnet-id subnet-id-public-subnet --allocation-
id eipalloc-id
```

c. （可选）如果您已经在第 5 步中为私有子网创建了路由表，请跳过这一步。否则，请使用下面的 create-route-table 命令为您的私有子网创建路由表。该命令将返回新路由表的 ID。

```
aws ec2 create-route-table --vpc-id vpc-1a2b3c4d5e6f1a2b3 --query
 RouteTable.RouteTableId --output text
```

d. 使用以下 create-route 命令，在私有子网的路由表中创建一条会将所有 IPv4 流量发送到 NAT 网关的路由。使用您在这一步或第 5 步中为私有子网创建的路由表的 ID。

```
aws ec2 create-route --route-table-id rtb-id-private --destination-cidr-
block 0.0.0.0/0 --gateway-id nat-id
```

e. （可选）如果您已在第 5 步中将路由表关联到私有子网，请跳过这一步。否则，请使用下面的 associate-route-table 命令将路由表关联到私有子网。使用您在这一步或第 5 步中为私有子网创建的路由表的 ID。

```
aws ec2 associate-route-table --route-table-id rtb-id-private --subnet-
id subnet-id-private-subnet
```

可视化 VPC 中的资源

本节旨在介绍如何使用资源地图选项卡查看 VPC 中资源的可视化表示。资源图中可以看到以下资源：

- VPC
- 子网
  - 可用区用字母表示。
  - 公有子网为绿色。
  - 私有子网为蓝色。
- 路由表

- Internet 网关
- 仅出口互联网网关
- NAT 网关
- 网关端点（Amazon S3 和 Amazon DynamoDB）

资源图会显示 VPC 内部资源之间的关系，以及流量如何从子网流向 NAT 网关、互联网网关和网关端点。

通过资源地图，您可以了解 VPC 的架构布局，查看子网数量、哪些子网与哪些路由表相关联以及哪些路由表具有通往 NAT 网关、互联网网关和网关端点的路由。

此外，您还可以通过资源地图发现不良或错误配置，例如与 NAT 网关断开连接的私有子网，或具有直接通往互联网网关的路由的私有子网。您可以在资源地图中选择路由表等资源，并编辑这些资源的配置。

可视化 VPC 中的资源

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 VPC。
3. 选择 VPC。
4. 选择资源地图选项卡以直观地显示资源。
5. 选择显示详细信息，以查看除默认显示的资源 ID 和区域以外的其他详细信息。

    - VPC：分配给 VPC 的 IPv4 和 IPv6 网址范围。
    - 子网：分配给每个子网的 IPv4 和 IPv6 CIDR 范围。
    - 路由表：子网关联和路由表中的路由数量。
    - 网络连接：与每种连接类型相关的详细信息：
        - 如果 VPC 中有公有子网，则存在互联网网关资源，其中包含路由数量以及使用互联网网关的流量的源子网和目标子网。
        - 如果有仅出口互联网网关，则存在仅出口互联网网关资源，其中包含路由数量以及使用仅出口互联网网关的流量的源子网和目标子网。
        - 如果有 NAT 网关，则存在 NAT 网关资源，其中包含 NAT 网关的网络接口数量以及弹性 IP 地址。
        - 如果有网关端点，则存在网关端点资源，其中包含您可以使用该端点连接的 AWS 服务（Amazon S3 或 Amazon DynamoDB）的名称。

6. 将鼠标指针悬停在资源上可查看资源之间的关系。实线表示资源之间的关系。虚线表示指向网络连接的网络流量。

将 CIDR 块添加到 VPC 或从中删除

本节旨在介绍如何将 IPv4 和 IPv6 CIDR 块添加到 VPC 或从中删除。

Important

- 默认情况下，您的 VPC 可以最多有 5 个 IPv4 CIDR 块和 5 个 IPv6 CIDR 块，但此限额可调整。有关更多信息，请参阅 Amazon VPC 配额。有关 VPC 的 CIDR 块限制的信息，请参阅 VPC CIDR 块。
- 如果您的 VPC 关联了多个 IPv4 CIDR 块，则可以取消一个 IPv4 CIDR 块与 VPC 的关联。您不能将主 IPv4 CIDR 块取消关联。您只能将整个 CIDR 块取消关联；不能将 CIDR 块的子集或 CIDR 块的合并范围取消关联。必须首先删除 CIDR 块中的所有子网。
- 如果不再需要在 VPC 中支持 IPv6，但需要继续使用 VPC 来创建 IPv4 资源并与之通信，则可以移除 IPv6 CIDR 块。
- 要移除 IPv6 CIDR 块，您必须首先将分配给子网中任何实例的任何 IPv6 地址取消分配。
- 移除 IPv6 CIDR 块关联不会自动删除您为 IPv6 网络配置的任何安全组规则、网络 ACL 规则或路由表路由。您必须手动修改或删除这些规则或路由。

使用控制台将 CIDR 块添加到 VPC 或从中删除

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Your VPCs（您的 VPC）。
3. 选择所需的 VPC，然后选择 Actions（操作）、Edit CIDRs（编辑 CIDR）。
4. 要删除 CIDR，选择 CIDR 旁的删除。
5. 要添加 CIDR，选择添加新的 IPv4 CIDR或添加新的 IPv6 CIDR。
6. 要为 IPv4 CIDR 块添加 CIDR，请执行以下操作之一：
   - 选择 IPv4 CIDR manual input（IPv4 CIDR 手动输入），然后输入 IPv4 CIDR 块。
   - 选择 IPAM-allocated IPv4 CIDR（IPAM 分配的 IPv4 CIDR），然后从 IPv4 IPAM 池中选择 CIDR。
   - 选择保存。

7. 要为 IPv6 CIDR 块添加 CIDR，请执行以下操作：

   - 如果使用 Amazon VPC IP 地址管理器，并且需要从 IPAM 池预置 IPv6 CIDR，则选择 IPAM 分配的 IPv6 CIDR 块。您可以通过两个选项，在 CIDR 块下为 VPC 预置一个 IP 地址范围：
     - 网络掩码长度：选择此选项可为 CIDR 选择网络掩码长度。请执行以下操作之一：
       - 如果已为 IPAM 池选择默认网络掩码长度，则可以选择默认为 IPAM 网络掩码长度，以使用 IPAM 管理员为 IPAM 池设置的默认网络掩码长度。有关可选默认网络掩码长度分配规则的更多信息，请参阅《Amazon VPC IPAM 用户指南》中的创建区域 IPv6 池。
       - 如果未为 IPAM 池选择默认网络掩码长度，则选择一个比 IPAM 池 CIDR 的网络掩码长度更具体的网络掩码长度。例如，假设 IPAM 池 CIDR 为 /50，则可以为 VPC 选择介于 /52 至 /60 之间的网络掩码长度。可能的网络掩码长度介于 /44 和 /60 之间，增量为 /4。
     - 选择 CIDR：选择此选项可手动输入 IPv6 地址。您只能选择比 IPAM 池 CIDR 的网络掩码长度更具体的网络掩码长度。例如，假设 IPAM 池 CIDR 为 /50，则可以为 VPC 选择介于 /52 至 /60 之间的网络掩码长度。可能的 IPv6 网络掩码长度介于 /44 和 /60 之间，增量为 /4。
   - 选择 Amazon 提供的 IPv6 CIDR 块，以从 Amazon 的 IPv6 地址池请求 IPv6 CIDR 块。对于 Network Border Group（网络边界组），选择 AWS 从中通告 IP 地址的组。Amazon 提供 /56 固定大小的 IPv6 CIDR 块。
   - 选择我拥有的 IPv6 CIDR，以预置您已经带到 AWS 的 IPv6 CIDR。有关更多信息，请参阅《Amazon EC2 用户指南》中的在 Amazon EC2 中使用您自己的 IP 地址（BYOIP）。您可以通过两个选项，在 CIDR 块下为 VPC 预置一个 IP 地址范围：
     - 无偏好：选择此选项使用 /56 的网络掩码长度。
     - 选择 CIDR：选择此选项可手动输入 IPv6 地址，然后选择比 BYOIP CIDR 的大小更具体的网络掩码长度。例如，假设 BYOIP 池 CIDR 为 /50，则可以为 VPC 选择介于 /52 至 /60 之间的网络掩码长度。可能的 IPv6 网络掩码长度介于 /44 和 /60 之间，增量为 /4。
     - 完成此操作后，选择选择 CIDR。
8. 选择关闭。
9. 将 CIDR 块添加到 VPC 后，您可以创建使用该新 CIDR 块的子网。有关更多信息，请参阅 创建子网。

使用 AWS CLI 将 CIDR 块与 VPC 关联或取消关联

使用 associate-vpc-cidr-block 和 disassociate-vpc-cidr-block 命令。

Amazon VPC 中的 DHCP 选项集

VPC 中的网络设备使用动态主机配置协议（DHCP）。您可以使用 DHCP 选项集控制虚拟网络中网络配置的以下方面：

- VPC 中设备使用的 DNS 服务器、域名或网络时间协议（NTP）服务器。
- 您的 VPC 是否启用了 DNS 解析。

内容

- 什么是 DHCP？
- DHCP 选项集概念
- 使用 DHCP 选项集

什么是 DHCP？

TCP/IP 网络上的每台设备都需要一个 IP 地址才能通过网络进行通信。过去，IP 地址必须手动为网络中的每台设备分配。如今，您可以由 DHCP 服务器使用动态主机配置协议（DHCP）来动态分配 IP 地址。

在 EC2 实例上运行的应用程序可以根据需要与 Amazon DHCP 服务器通信，以检索其 IP 地址租约或其他网络配置信息（例如 Amazon DNS 服务器的 IP 地址或 VPC 中路由器的 IP 地址）。

您可以通过使用 DHCP 选项集指定 Amazon DHCP 服务器提供的网络配置。

如果 VPC 配置要求应用程序直接向 Amazon IPv6 DHCP 服务器发出请求，请注意以下各项：

- 双堆栈子网中的 EC2 实例只能从 IPv6 DHCP 服务器检索其 IPv6 地址。它无法从 IPv6 DHCP 服务器检索任何其他联网配置，例如 DNS 服务器名称或域名。
- 仅使用 IPv6 的子网中的 EC2 实例可以从 IPv6 DHCP 服务器检索其 IPv6 地址，并可以检索其他联网配置信息，例如 DNS 服务器名称和域名。
- 对于仅限 IPv6 子网中的 EC2 实例，如果 DHCP 选项集中明确提及“AmazonProvidedDNS”，则 IPv4 DHCP Server 将返回 169.254.169.253 作为域名服务器。如果选项集中缺少“AmazonProvidedDNS”，则无论选项集中是否提及其他 IPv4 域名服务器，IPv4 DHCP Server 都不会返回地址。

Amazon DHCP 服务器还可以使用前缀委派为 VPC 中的网络接口提供完整的 IPv4 或 IPv6 前缀（请参阅《Amazon EC2 用户指南》中的为 Amazon EC2 网络接口分配前缀）。DHCP 响应中不提供 IPv4 前缀委派。可以使用 IMDS 检索分配给接口的 IPv4 前缀（请参阅《Amazon EC2 用户指南》中的实例元数据类别)。

DHCP 选项集概念

DHCP 选项集是 VPC 中的资源实例使用的一组网络设置（例如 EC2 实例），以用于通过您的虚拟网络进行通信。

每个区域都有默认的 DHCP 选项集。除非您创建自定义 DHCP 选项集并将其与 VPC 关联，或者在未没有 DHCP 选项集的情况下配置 VPC，否则每个 VPC 都会使用其区域的默认 DHCP 选项集。

如果您的 VPC 未配置 DHCP 选项集：

- 对基于 Nitro System 构建的 EC2 实例，AWS 将 `169.254.169.253` 配置为默认域名服务器。
- 对基于 Xen 构建的 EC2 实例，将不配置域名服务器，并且由于 VPC 中的实例无法访问 DNS 服务器，这些实例将无法访问互联网。

您可以有将一个 DHCP 选项集与多个 VPC 关联，但每个 VPC 只能有一个关联的 DHCP 选项集。

如果您删除一个 VPC，与该 VPC 关联的 DHCP 选项集将与该 VPC 解除关联。

内容

- 默认 DHCP 选项集
- 自定义 DHCP 选项集

默认 DHCP 选项集

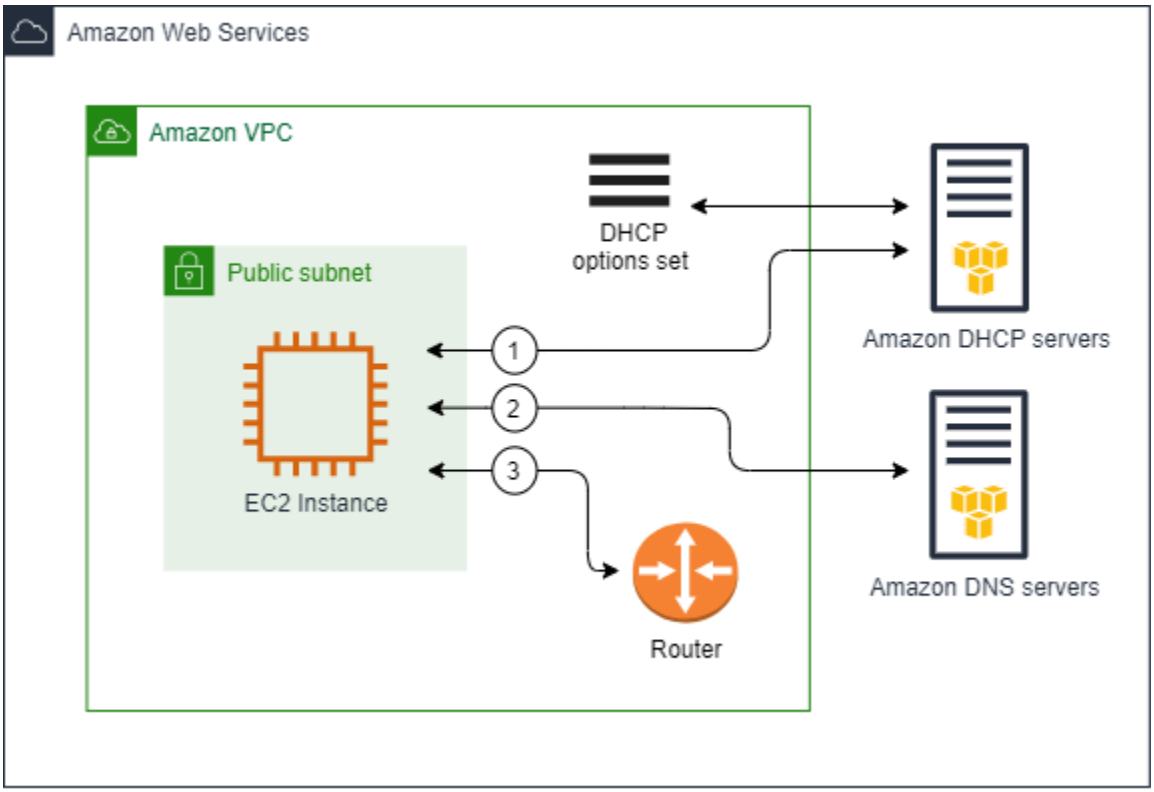
默认 DHCP 选项集包含了以下设置：

- 域名服务器：供网络接口用于域名解析的 DNS 服务器。对于默认 DHCP 选项集，这始终是 `AmazonProvidedDNS`。有关更多信息，请参阅 Amazon DNS 服务器。
- 域名：客户端通过域名系统（DNS）解析主机名时应使用的域名。有关用于 EC2 实例的域名的更多信息，请参阅 Amazon EC2 实例主机名。
- IPv6 首选租赁时间：为其分配了 IPv6 的正在运行的实例续订 DHCPv6 租约的频率。默认租赁时间为 140 秒。通常在租赁时间已过一半时进行租约续订。

当您使用默认的 DHCP 选项集时，不会使用以下设置，但是 EC2 实例有默认设置：

- NTP 服务器：默认情况下，EC2 实例使用 Amazon Time Sync Service 检索时间。
- NetBIOS 名称服务器：对于运行 Windows 的 EC2 实例，NetBIOS 电脑名称是分配给实例的一个友好名称，用于在网络上识别它。NetBIOS 名称服务器负责维护 NetBIOS 电脑名称与使用 NetBIOS 作为命名服务的网络的网络地址之间的映射列表。
- NetBIOS 节点类型：对于运行 Windows 的 EC2 实例，这是这些实例用于将 NetBIOS 名称解析为 IP 地址的方法。

如果使用原定设置的选项集，Amazon DHCP 服务器将使用原定设置选项集中的网络设置。当您在 VPC 中启动实例时，他们将执行下图中显示的操作：(1) 与 DHCP 服务器交互，(2) 与 Amazon DNS 服务器交互，(3) 然后通过 VPC 的路由器连接到网络中的其他设备。这些实例可以随时与 Amazon DHCP 服务器进行交互，以获取其 IP 地址租赁和其他网络设置。



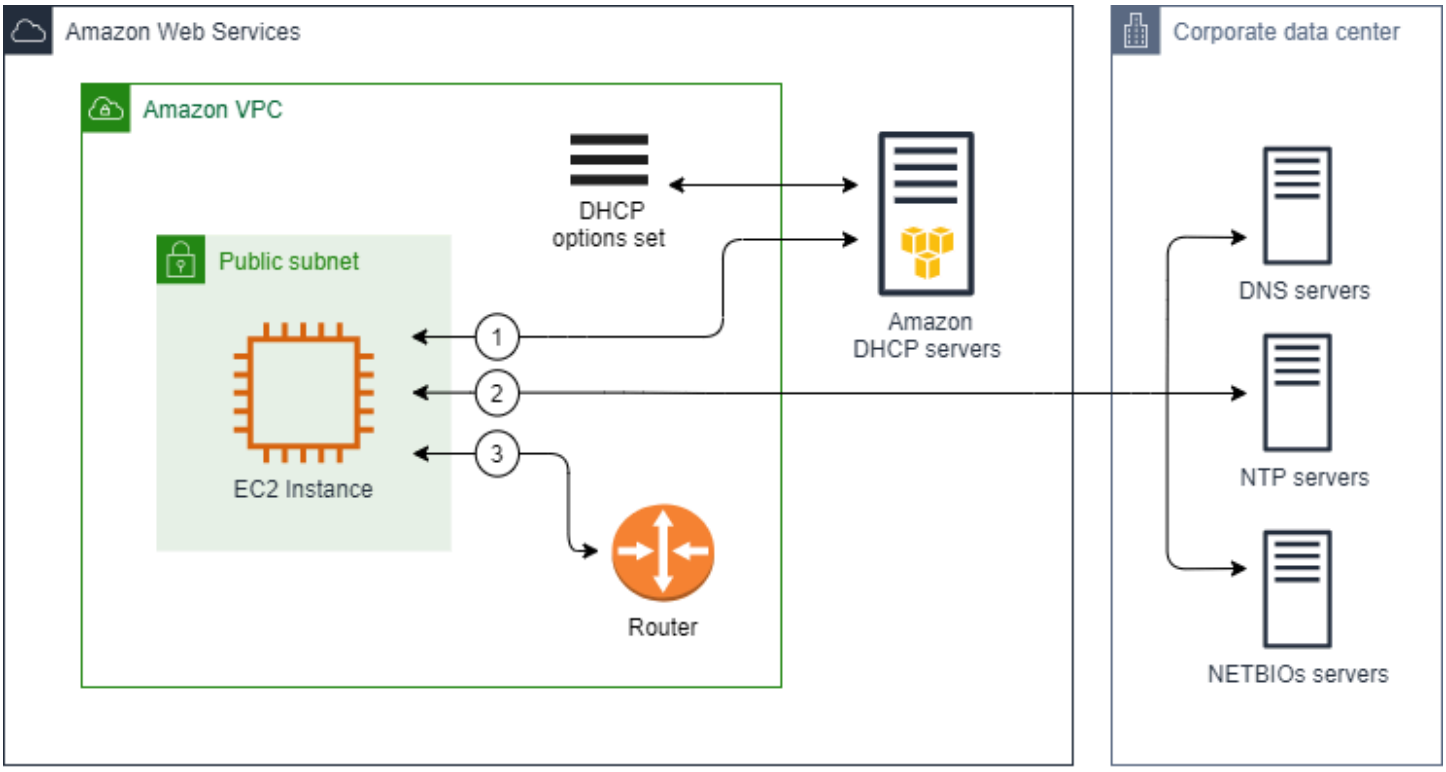
自定义 DHCP 选项集

您可以使用以下设置创建自定义 DHCP 选项集，然后将此选项集与 VPC 关联：

- 域名服务器：供网络接口用于域名解析的 DNS 服务器。
- 域名：客户端通过域名系统（DNS）解析主机名时应使用的域名。

- NTP 服务器：为实例提供时间的 NTP 服务器。
- NetBIOS 名称服务器：对于运行 Windows 的 EC2 实例，NetBIOS 电脑名称是分配给实例的一个友好名称，用于在网络上识别它。NetBIOS 名称服务器负责维护 NetBIOS 电脑名称与使用 NetBIOS 作为命名服务的网络的网络地址之间的映射列表。
- NetBIOS 节点类型：对于运行 Windows 的 EC2 实例，即这些实例用于将 NetBIOS 名称解析为 IP 地址的方法。
- IPv6 首选租赁时间（可选）：为其分配了 IPv6 的正在运行的实例续订 DHCPv6 租约的频率值（以秒、分钟、小时或年为单位）。可接受的值介于 140 到 4294967295 秒（大约 138 年）之间。如果未输入值，则默租赁时间为 140 秒。如果您对 EC2 实例使用长期寻址，则可以增加租赁时间，避免频繁的租约续订请求。通常在租赁时间已过一半时进行租约续订。

如果您使用自定义选项集，在 VPC 中启动的实例将执行如图所示的以下操作：(1) 使用自定义 DHCP 选项集中的网络设置，(2) 与自定义 DHCP 选项集中指定的 DNS、NTP 和 NetBIOS 服务器进行交互，然后 (3) 通过 VPC 的路由器连接到网络中的其他设备。



相关任务

- 创建 DHCP 选项集
- 更改与 VPC 关联的选项集。

使用 DHCP 选项集

使用以下过程查看 DHCP 选项集，并进行使用。有关 DHCP 选项集工作原理的更多信息，请参阅 the section called “DHCP 选项集概念”。

任务

- 创建 DHCP 选项集
- 更改与 VPC 关联的选项集。
- 删除 DHCP 选项集

创建 DHCP 选项集

借助自定义 DHCP 选项集，您可以使用自己的 DNS 服务器、域名等来自定义 VPC。您可以根据需要，任意创建额外 DHCP 选项集。但是，您一次只能将一个 VPC 与一个 DHCP 选项集相关联。

> **Note**
>
> 在您创建 DHCP 选项集之后，您便无法再对其进行修改。要为您的 VPC 更新 DHCP 选项，您必须创建新的 DHCP 选项集，然后将其关联到您的 VPC。

要使用控制台创建 DHCP 选项集

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 DHCP option sets（DHCP 选项集）。
3. 选择 Create DHCP options set（创建 DHCP 选项集）。
4. 对于 Tag settings（标签设置），根据需要输入 DHCP 选项集的名称。如果输入一个值，将会自动为 DHCP 选项集创建一个名称标签。
5. 对于DHCP 选项，提供所需的配置参数。
   - Domain name（域名，可选）：输入客户端通过域名系统解析主机名时应使用的域名。如果您未使用 AmazonProvidedDNS，您的自定义域名服务器必须正确解析主机名。如果您使用 Amazon Route 53 私有托管区域，则可以使用 AmazonProvidedDNS。有关更多信息，请参阅 VPC 中的 DNS 属性。

> **Note**
>
> 仅使用您完全控制的域名。

某些 Linux 操作系统接受由空格分隔的多个域名。但是，Windows 以及其他 Linux 操作系统将该值视为单个域，因而会导致意外行为。如果您的 DHCP 选项集与其中实例所运行操作系统将该值视为单个域的 VPC 关联，请仅指定一个域名。

- Domain name servers（域名服务器，可选）：输入将用于将主机名称解析为主机 IP 地址的 DNS 服务器。

  您可以输入 **AmazonProvidedDNS** 或自定义域名服务器。使用两者都可能会导致意外行为。您最多可以输入四个 IPv4 域名服务器（或最多三个 IPv4 域名服务器和 **AmazonProvidedDNS**）和四个 IPv6 域名服务器的 IP 地址，用逗号分隔。尽管最多可以指定八个域名服务器，但某些操作系统可能会施加较低的限制。有关 AmazonProvidedDNS 和 Amazon DNS 服务器的更多信息，请参阅 Amazon DNS 服务器。

  > **Important**
  >
  > 如果您的 VPC 有互联网网关，确保指定您自己的 DNS 服务器或 Amazon DNS 服务器 (AmazonProvidedDNS) 作为域名服务器值。否则，VPC 中的实例将无法访问 DNS，这样会禁用互联网访问。

- NTP servers（NTP 服务器，可选）：输入最多八个网络时间协议（NTP）服务器的 IP 地址（四个 IPv4 地址和四个 IPv6 地址）。

  NTP 服务器为您的网络提供时间。您可以在 IPv4 地址 `169.254.169.123` 或 IPv6 地址 `fd00:ec2::123` 指定 Amazon Time Sync Service。原定设置下，实例与 Amazon Time Sync Service 通信。请注意，IPv6 地址只能在基于 Nitro 系统构建的 EC2 实例上访问。

  有关 NTP 服务器选项的更多信息，请参阅 RFC 2132。有关 Amazon Time Sync Service 的更多信息，请参阅《Amazon EC2 用户指南》中的为您的实例设置时间。

- NetBIOS name servers（NetBIOS 名称服务器，可选）：输入最多四个 NetBIOS 名称服务器的 IP 地址。

对于运行 Windows 操作系统的 EC2 实例，NetBIOS 电脑名称是分配给实例的一个友好名称，用于在网络上识别它。NetBIOS 名称服务器负责维护 NetBIOS 电脑名称与使用 NetBIOS 作为命名服务的网络的网络地址之间的映射列表。

- NetBIOS node type（NetBIOS 节点类型，可选）：输入 **1**、**2**、**4** 或 **8**。我们建议您指定 **2**（点对点或 P 节点）。目前不支持广播和多播。有关这些节点类型的更多信息，请参阅 RFC 2132 的第 8.7 节，以及 RFC1001 的第 10 节。

  对于运行 Windows 操作系统的 EC2 实例，这是这些实例用于将 NetBIOS 名称解析为 IP 地址的方法。在原定设置选项集中，NetBIOS 节点类型没有值。

- IPv6 首选租赁时间（可选）：为其分配了 IPv6 的正在运行的实例续订 DHCPv6 租约的频率值（以秒、分钟、小时或年为单位）。可接受的值介于 140 到 2147483647 秒（大约 68 年）之间。如果未输入值，则默租赁时间为 140 秒。如果您对 EC2 实例使用长期寻址，则可以增加租赁时间，避免频繁的租约续订请求。通常在租赁时间已过一半时进行租约续订。

6. 添加 Tags（标签）。
7. 选择 Create DHCP options set（创建 DHCP 选项集）。将新 DHCP 选项集的名称或 ID 记录下来。
8. 要配置您的 VPC 以使用新的选项集，请参阅 更改与 VPC 关联的选项集。。

要使用命令行为您的 VPC 创建 DHCP 选项集

- create-dhcp-options (AWS CLI)
- New-EC2DhcpOption (AWS Tools for Windows PowerShell)

更改与 VPC 关联的选项集。

创建 DHCP 选项集之后，您可以将其与一个或多个 VPC 关联。您一次只能将一个 DHCP 选项集与一个 VPC 相关联。如果您未将 DHCP 选项集与 VPC 关联，则这样会禁用 VPC 中的域名解析。

在您将新的 DHCP 选项集与 VPC 关联时，任何现有实例以及您在 VPC 内启动的所有新实例都将使用新选项。无需重新开始或重新启动您的实例。根据实例更新 DHCP 租赁权的频率，实例会在几个小时内自动拾取更改。如果您愿意，您也可以使用实例上的操作系统，直接更新租赁权。

要使用控制台更改与 VPC 相关联的 DHCP 选项集

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Your VPCs(您的 VPC)。

3. 选择 VPC 的复选框，然后依次选择 Actions（操作）、Edit VPC settings（编辑 VPC 设置）。
4. 对于 DHCP options set（DHCP 选项集），选择新的 DHCP 选项集。或者，选择没有 DHCP 选项集以禁用 VPC 的域名解析。
5. 选择保存。

要使用命令行更改与 VPC 相关联的 DHCP 选项集

- associate-dhcp-options (AWS CLI)
- Register-EC2DhcpOption (AWS Tools for Windows PowerShell)

删除 DHCP 选项集

当您不再需要 DHCP 选项集时，您可以使按照以下步骤删除 DHCP 选项集。如果正在使用 DHCP 选项集，则无法将其删除。对于要删除的每个与 DHCP 选项集关联的 VPC，您必须将不同的 DHCP 选项集与该 VPC 关联或将 VPC 配置为不使用 DHCP 选项集。有关更多信息，请参阅 the section called “更改与 VPC 关联的选项集。”。

要使用控制台删除 DHCP 选项集

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 DHCP option sets（DHCP 选项集）。
3. 选择 DHCP 选项集的单选按钮，然后依次选择操作、删除 DHCP 选项集。
4. 提示进行确认时，输入 **`delete`**，然后选择删除 DHCP 选项集。

要使用命令行删除 DHCP 选项集

- delete-dhcp-options (AWS CLI)
- Remove-EC2DhcpOption (AWS Tools for Windows PowerShell)

VPC 中的 DNS 属性

域名系统 (DNS) 是 Internet 中名称使用的标准，以将名称解析到各自相应的 IP 地址。DNS 主机名是可以唯一并绝对区分计算机的名称；它由主机名和域名组成。DNS 服务器会将 DNS 主机名称解析到其相应的 IP 地址。

公有 IPv4 地址可实现 Internet 间的通信，而私有 IPv4 地址可实现实例网络内的通信。有关更多信息，请参阅 为 VPC 和子网分配 IP 地址。

Amazon 为您的 VPC 提供 DNS 服务器（Amazon Route 53 Resolver）。要使用您自己的 DNS 服务器，请为您的 VPC 创建一组新的 DHCP 选项。有关更多信息，请参阅 Amazon VPC 中的 DHCP 选项集。

内容

- 了解 Amazon DNS
- 查看您的 EC2 实例的 DNS 主机名称
- 查看和更新 VPC 的 DNS 属性

了解 Amazon DNS

身为 AWS 架构师或管理员，您会遇到的一种基本联网组件是 Amazon DNS 服务器，也称为 Route 53 Resolver。此 DNS 解析器服务原生集成到您所在 AWS 区域的每个可用区中，可为虚拟私有云（VPC）中的域名解析提供可靠且可扩展的解决方案。在本节中，您会了解到 Amazon DNS 服务器的 IP 地址、Amazon DNS 服务器可以解析的私有 DNS 主机名以及管理着 Amazon DNS 服务器使用的规则。

内容

- Amazon DNS 服务器
- 规则和注意事项
- EC2 实例的 DNS 主机名称
- VPC 中的 DNS 属性
- DNS 配额
- 私有托管区域

Amazon DNS 服务器

Route 53 Resolver（也称“Amazon DNS 服务器”或“AmazonProvidedDNS”）是一种 DNS 解析程序服务，内置于 AWS 区域内的每个可用区中。Route 53 Resolver 位于 `169.254.169.253`（IPv4）、`fd00:ec2::253`（IPv6）以及预置到“VPC+2”的主要私有 IPV4 CIDR 范围。例如，如果您 VPC 的 IPv4 CIDR 为 `10.0.0.0/16`、IPv6 CIDR 为 `2001:db8::/32`，则可通过 `169.254.169.253`（IPv4）、`fd00:ec2::253`（IPv6）或 `10.0.0.2`（IPv4）访问 Route

53 Resolver。VPC 内的资源使用链路本地地址进行 DNS 查询。这些查询会私下传输到 Route 53 Resolver，但在网络上不可见。在仅限 IPv6 子网中，只要“AmazonProvidedDNS”是 DHCP 选项集中的域名服务器，便仍可访问 IPv4 链路本地地址（169.254.169.253）。

当您将实例启动到 VPC 中时，我们会为该实例提供一个私有 DNS 主机名。如果该实例配置了一个公有 IPv4 地址并且启用了 VPC DNS 属性，我们还会提供一个公有 DNS 主机名。

私有 DNS 主机名的格式取决于您在启动 EC2 实例时如何配置它。有关私有 DNS 主机名类型的更多信息，请参阅《Amazon EC2 用户指南》中的 Amazon EC2 实例主机名类型。

您的 VPC 中的 Amazon DNS 服务器用于解析您在 Route 53 中的私有托管区域中指定的 DNS 域名。有关私有托管区域的更多信息，请参阅 Amazon Route 53 开发人员指南 中的使用私有托管区域。

规则和注意事项

使用 Amazon DNS 服务器时，适用以下规则和注意事项。

- 您无法使用网络 ACL 或安全组筛选进出 Amazon DNS 服务器的流量。
- 使用 Hadoop 框架的服务（如 Amazon EMR）要求实例解析自己的完全限定域名 (FQDN)。这种情况下，如果 `domain-name-servers` 选项设置为自定义值，则 DNS 解析可能会失败。要确保正确解析 DNS，请考虑在您的 DNS 服务器添加条件转发服务器，将针对域 *`region-name`*`.compute.internal` 的查询转发到 Amazon DNS 服务器。有关更多信息，请参阅 Amazon EMR 管理指南中的设置 VPC 以托管集群。
- Amazon Route 53 Resolver 只支持递归 DNS 查询。

EC2 实例的 DNS 主机名称

当您启动实例时，实例始终会收到一个私有 IPv4 地址和一个与其私有 IPv4 地址对应的私有 DNS 主机名。如果您的实例具有公有 IPv4 地址，则该实例 VPC 的 DNS 属性决定实例是否接收与公有 IPv4 地址对应的公有 DNS 主机名。有关更多信息，请参阅 VPC 中的 DNS 属性。

启用 Amazon 提供的 DNS 服务器后，DNS 主机名按如下方式解析。

私有 IPv4 DNS 名称

实例的私有 IPv4 DNS 主机名解析为私有 IPv4 地址。您可以使用私有 IPv4 DNS 主机名，在同一 VPC 中或连接的 VPC 中的实例之间进行通信。有关更多信息，请参阅《Amazon EC2 用户指南》中的私有 IPv4 地址。

公有 IPv4 DNS 名称

实例的公有 IPv4 DNS 主机名解析为（该实例网络外）的公有 IPv4 地址或（该实例网络内）的私有 IPv4 地址。有关更多信息，请参阅《Amazon EC2 用户指南》中的公有 IPv4 地址。

要通过 VPC 对等连接将公有 IPv4 DNS 名称解析为私有 IPv4 地址，必须启用对等连接的 DNS 解析功能。有关更多信息，请参阅实现对 VPC 对等连接的 DNS 解析。

私有资源 DNS 名称

基于 RBN 的 DNS 名称，它可以解析为此实例选择的 A 和 AAAA DNS 记录。此 DNS 主机名在双堆栈和仅 IPv6 子网中的实例的实例详细信息中可见。有关 RBN 的更多信息，请参阅《Amazon EC2 用户指南》中的 EC2 实例主机名类型。

VPC 中的 DNS 属性

以下 VPC 属性决定了为您的 VPC 提供的 DNS 支持。如果两项属性均启用，则如果在创建时为启动到 VPC 内的实例分配了公有 IPv4 地址或弹性 IP 地址，则该实例会接收公有 DNS 主机名。如果您为之前未启用两项属性的 VPC 启用这两项属性，则已经启动至该 VPC 的实例将接收公有 DNS 主机名（如果它们具有公有 IPv4 地址或弹性 IP 地址）。

要检查是否为 VPC 启用了这些属性，请参阅查看和更新 VPC 的 DNS 属性。

属性	描述
`enableDnsHostnames`	确定 VPC 是否支持将公有 DNS 主机名分配给具有公有 IP 地址的实例。 除非 VPC 是默认 VPC，否则此属性的默认值为 `false`。请注意下面此属性的规则和注意事项。
`enableDnsSupport`	确定 VPC 是否支持通过 Amazon 提供的 DNS 服务器进行 DNS 解析。 如果此属性是 `true`，对 Amazon 提供的 DNS 服务器的查询成功。有关更多信息，请参阅 Amazon DNS 服务器。 此属性的默认值为 `true`。请注意下面此属性的规则和注意事项。

规则和注意事项

- 如果两个属性都设置为 `true`，则会发生以下情况：

  - 具有公有 IP 地址的实例会收到对应的公有 DNS 主机名。
  - Amazon Route 53 Resolver 服务器可以解析 Amazon 提供的私有 DNS 主机名。
- 如果至少将某个属性设置为 `false`，将出现以下情况：
  - 具有公有 IP 地址的实例不会收到对应的公有 DNS 主机名。
  - Amazon Route 53 Resolver 无法解析 Amazon 提供的私有 DNS 主机名。
  - 如果 DHCP 选项集中存在自定义域名，则实例会收到自定义私有 DNS 主机名。如果您未使用 Amazon Route 53 Resolver 服务器，您的自定义域名服务器必须正确解析主机名。
- 如果您使用在 Amazon Route 53 中的私有托管区域中定义的自定义 DNS 域名，或者使用具有接口 VPC 端点的私有 DNS (AWS PrivateLink)，则必须将 `enableDnsHostnames` 和 `enableDnsSupport` 属性设置为 `true`。
- Amazon Route 53 Resolver 可以将私有 DNS 主机名解析为全部地址空间内的私有 IPv4 地址，包括您的 VPC 的 IPv4 地址范围不在 RFC 1918 指定的私有 IPv4 地址范围内的情况。但是，如果您在 2016 年 10 月之前创建了 VPC，并且您的 VPC 的 IPv4 地址范围不在这些地址范围内，则 Amazon Route 53 Resolver将无法解析私有 DNS 主机名。要支持这种情况，请联系 支持。

DNS 配额

对于使用本地链路地址的服务，每秒数据包数（PPS）限制为 1024 个。此限制是 Route 53 Resolver DNS 查询、实例元数据服务（IMDS）请求、Amazon Time Service 网络时间协议（NTP）请求和 Windows 许可服务（适用于基于 Microsoft Windows 的实例）请求的总和。无法提高此配额。

由 Route 53 Resolver 支持的每秒 DNS 查询数量因查询类型、响应大小和所用协议而异。有关可扩展 DNS 架构的更多信息和建议，请参阅具有 Active Directory 的 AWS 混合 DNS 技术指南。

如果您达到配额，Route 53 Resolver 将拒绝流量。达到配额的部分原因可能是 DNS 节流问题，或者是使用 Route 53 Resolver 网络接口的实例元数据查询。有关如何解决 VPC DNS 节流问题的信息，请参阅如何确定我对 Amazon 提供的 DNS 服务器的 DNS 查询是否由于 VPC DNS 节流而失败。有关实例元数据检索的更多信息，请参阅《Amazon EC2 用户指南》中的检索实例元数据。

私有托管区域

要使用自定义 DNS 域名（如 `example.com`）而不是使用私有 IPv4 地址或AWS提供的私有 DNS 主机名来访问您的 VPC 中的资源，您可以在 Route 53 中创建一个私有托管区域。私有托管区域就是一个容器，其中包含的信息说明您希望如何在一个或多个 VPC 中为某个域及其子域路由流量而不将您的资源公开到 Internet。您可以创建 Route 53 资源记录集，用来确定 Route 53 将如何响应对您的域及其子域的查询。例如，如果您希望将对 example.com 的浏览器请求路由到您 VPC 中的某个 Web 服务

器，可以在您的私有托管区域中创建一条 A 记录并指定该 Web 服务器的 IP 地址。有关创建私有托管区域的更多信息，请参阅 Amazon Route 53 开发人员指南 中的使用私有托管区域。

要使用自定义 DNS 域名访问资源，必须连接到您的 VPC 中的实例。在您的实例中，您可通过使用 `ping` 命令来测试是否可从私有托管区域中的资源的自定义 DNS 名称访问该资源；例如，`ping mywebserver.example.com`。(您必须确保您的实例的安全组规则允许入站 ICMP 流量才能使 `ping` 命令正常运行。)

私有托管区域不支持 VPC 外的传递关系；例如，您不能使用资源的自定义私有 DNS 名称从 VPN 连接的另一端访问资源。

> **Important**
>
> 如果您使用在 Amazon Route 53 中的私有托管区域中定义的自定义 DNS 域名，则必须将 `enableDnsHostnames` 和 `enableDnsSupport` 属性设置为 `true`。

查看您的 EC2 实例的 DNS 主机名称

您可以使用 Amazon EC2 控制台或命令行查看运行实例或网络接口的 DNS 主机名。知道这些主机名对于连接到您的资源非常重要。

为与实例关联的 VPC 启用了 DNS 选项时，公有 DNS (IPv4) 和私有 DNS 字段可用。有关更多信息，请参阅 the section called “VPC 中的 DNS 属性”。

实例

使用控制台查看实例的 DNS 主机名称

1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 在导航窗格中，选择实例。
3. 从列表中选择您的 实例。
4. 在详细信息窗格中，Public DNS (IPv4) 和 Private DNS 字段显示 DNS 主机名 (如适用)。

使用命令行查看实例的 DNS 主机名

- describe-instances（AWS CLI）
- Get-EC2InstanceAWS Tools for Windows PowerShell

网络接口

使用控制台查看网络接口的私有 DNS 主机名

1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 在导航窗格中，选择 Network Interfaces。
3. 从列表中选择网络接口。
4. 在详细信息窗格中，私有 DNS (IPv4) 字段显示私有 DNS 主机名。

使用命令行查看网络接口的 DNS 主机名

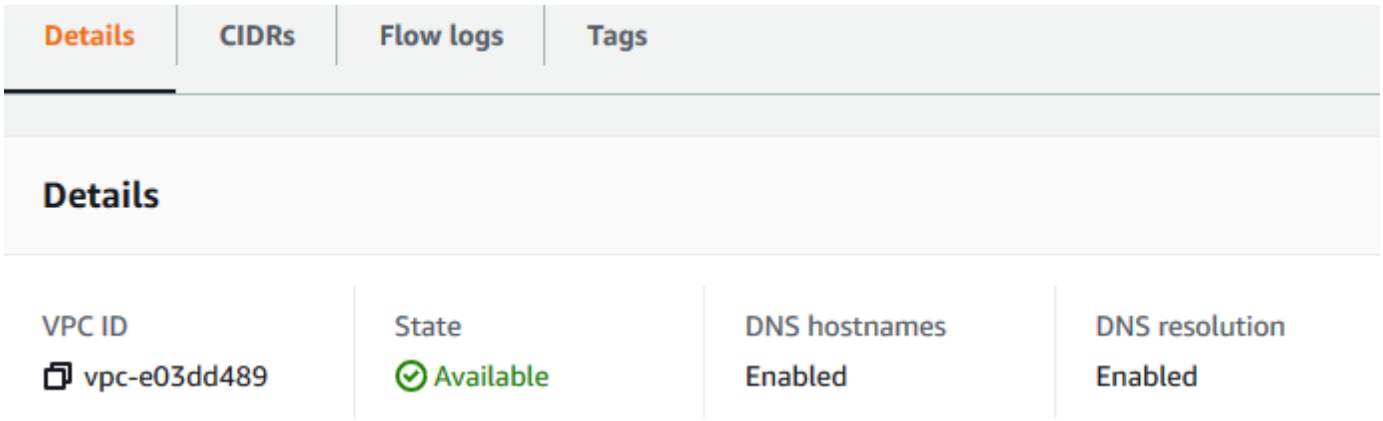
- describe-network-interfaces (AWS CLI)
- Get-EC2NetworkInterface (AWS Tools for Windows PowerShell)

查看和更新 VPC 的 DNS 属性

您可以通过 Amazon VPC 控制台查看并更新您的 VPC 中的 DNS 支持属性。这些设置控制着实例能否获得公有 DNS 主机名以及 Amazon DNS 服务器能否解析私有 DNS 名称。正确配置这些属性对于确保 VPC 内部的无缝通信至关重要。

使用控制台描述和更新 VPC 的 DNS 支持

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Your VPCs(您的 VPC)。
3. 选中该 VPC 的复选框。
4. 查看 Details (详细信息) 中的信息。在此示例中，已同时启用 DNS hostnames (DNS 主机名) 和 DNS resolution (DNS 解析)。



5. 要更新这些设置，请选择 Actions（操作），然后选择 Edit VPC settings（编辑 VPC 设置）。在相应的 DNS 属性上选择或清除 Enable（启用），然后选择 Save changes（保存更改）。

使用命令行说明 VPC 的 DNS 支持

- describe-vpc-attribute (AWS CLI)
- Get-EC2VpcAttribute (AWS Tools for Windows PowerShell)

使用命令行更新 VPC 的 DNS 支持

- modify-vpc-attribute (AWS CLI)
- Edit-EC2VpcAttribute (AWS Tools for Windows PowerShell)

VPC 的网络地址用量

网络地址用量（NAU）是应用于虚拟网络中资源的指标，可以帮助您规划和监控 VPC 的大小。每个 NAU 单元都会计入表示 VPC 大小的总数。

务必了解构成 VPC 的 NAU 单元总数，因为以下 VPC 限额会限制 VPC 的大小：

- 网络地址用量 – 单个 VPC 可以拥有的最大 NAU 单元数。默认情况下，每个 VPC 最多可以有 64,000 个 NAU 单元。您可以请求将限额提高到 256,000。
- 对等网络地址用量 – VPC 及其所有对等 VPC 的最大 NAU 单元数。如果一个 VPC 与同一区域中的其他 VPC 对等，则在默认情况下，组合的 VPC 最多可以有 128,000 个 NAU 单元。您可以请求将限额提高到 512,000。跨不同区域的对等 VPC 不会影响此限制。

您可以通过以下方式使用 NAU：

- 在创建虚拟网络之前，计算 NAU 单元以帮助您决定是否应将工作负载分散到多个 VPC 上。
- 创建 VPC 后，使用 Amazon CloudWatch 监控 VPC 的 NAU 用量，避免其超过 NAU 限额限制。有关更多信息，请参阅 the section called “CloudWatch 指标”。

NAU 的计算方式

了解 NAU 的计算方式可以帮助您进行 VPC 扩缩计划。

下表说明了构成 VPC 中 NAU 计数的资源，以及每种资源所使用的 NAU 单元数。部分 AWS 资源表示为单个 NAU 单元，部分资源表示为多个 NAU 单元。您可以使用该表来了解 NAU 的计算方式。

资源	NAU 单元
分配给 VPC 中 EC2 实例网络接口的每个私有或公有 IPv4 地址以及每个 IPv6 地址	1
附加到 EC2 实例的其他网络接口	1
分配给网络接口的前缀	1
每个 AZ 的网络负载均衡器	6
每个可用区的网关负载均衡器	6
每个 AZ 的 VPC 端点	6
中转网关挂载	6
Lambda 函数	6
NAT 网关	6
EFS 挂载目标	6
EFA 接口（带 ENA 设备的 EFA）或仅 EFA 的接口	1
Amazon EKS 容器组（pod）	1

NAU 示例

以下示例说明了 NAU 的计算方式。

示例 1 – 使用 VPC 对等连接的两个 VPC

同一区域中的对等 VPC 会占用 NAU 组合限额。

- VPC 1
  - 2 个子网分布于独立的可用区内，有 50 个网络负载均衡器 – 600 个 NAU 单元

  - 一个子网中的 5,000 个实例（每个实例都具有一 IPv4 地址和 IPv6 地址），另一个子网中的 5,000 个实例（每个实例都具有 IPv4 地址和 IPv6 地址）- 20,000 个单位
  - 100 个 Lambda 函数 – 600 个 NAU 单元
- VPC 2
  - 2 个子网分布于独立的可用区内，有 50 个网络负载均衡器 – 600 个 NAU 单元
  - 一个子网中的 5,000 个实例（每个实例都具有一 IPv4 地址和 IPv6 地址），另一个子网中的 5,000 个实例（每个实例都具有 IPv4 地址和 IPv6 地址）- 20,000 个单位
  - 100 个 Lambda 函数 – 600 个 NAU 单元
- 对等 NAU 总数：42,400 个单元
- 对等 NAU 默认限额：128,000 个单元

示例 2 – 使用中转网关连接的两个 VPC

与对等 VPC 不同，使用中转网关连接的 VPC 不会占用 NAU 组合限额。

- VPC 1
  - 2 个子网分布于独立的可用区内，有 50 个网络负载均衡器 – 600 个 NAU 单元
  - 一个子网中的 5,000 个实例（每个实例都具有一 IPv4 地址和 IPv6 地址），另一个子网中的 5,000 个实例（每个实例都具有 IPv4 地址和 IPv6 地址）- 20,000 个单位
  - 100 个 Lambda 函数 – 600 个 NAU 单元
- VPC 2
  - 2 个子网分布于独立的可用区内，有 50 个网络负载均衡器 – 600 个 NAU 单元
  - 一个子网中的 5,000 个实例（每个实例都具有一 IPv4 地址和 IPv6 地址），另一个子网中的 5,000 个实例（每个实例都具有 IPv4 地址和 IPv6 地址）- 20,000 个单位
  - 100 个 Lambda 函数 – 600 个 NAU 单元
- 每个 VPC 的 NAU 总数：21,200 个单元
- 每个 VPC 的 NAU 默认限额：64,000 个单元

与其他账户共享 VPC 子网

VPC 子网共享允许多个 AWS 账户将其应用程序资源 [例如 Amazon EC2 实例、Amazon Relational Database Service（RDS）数据库、Amazon Redshift 集群和 AWS Lambda 函数] 创建到共享的集中管理式 Virtual Private Cloud（VPC）中。在此模型中，拥有 VPC 的账户（拥有者）与属于 AWS

Organizations 中同一企业的其他账户（参与者）共享一个或多个子网。共享子网之后，参与者可以查看、创建、修改和删除与他们共享的子网中的应用程序资源。参与者无法查看、修改或删除属于其他参与者或 VPC 拥有者的资源。

您可以共享您的 VPC 子网，以针对需要高度互连且位于相同的信任边界内的应用程序，利用 VPC 内的隐式路由。这可减少您创建和管理的 VPC 数量，同时使用单独的账户进行计费和访问控制。您可以通过使用连接功能（例如 AWS PrivateLink、中转网关和 VPC 对等连接）互连共享的 Amazon VPC 子网来进一步简化网络拓扑。有关 VPC 子网共享优点的更多信息，请参阅 VPC 共享：多账户和 VPC 管理的新方法。

VPC 子网共享存在相关的限额。有关更多信息，请参阅 VPC 子网共享。

内容

- 共享子网的先决条件
- 使用共享子网
- 拥有者与参与者的计费和计量
- 所有者和参与者的责任和权限
- AWS 资源和共享 VPC 子网

共享子网的先决条件

本节旨在介绍使用共享子网的先决条件：

- VPC 所有者和参与者的账户必须由 AWS Organizations 管理。
- 您必须从组织的管理账户在 AWS RAM 控制台中启用资源共享。有关更多信息，请参阅《AWS RAM 用户指南》中的允许在 AWS Organizations 内共享资源。
- 您必须创建一个资源共享。您可以在创建资源共享时指定要共享的子网，也可以稍后使用下一节中的过程将子网添加到资源共享中。有关更多信息，请参阅《AWS RAM 用户指南》中的 Create a resource share。

使用共享子网

本节旨在介绍如何在 AWS 控制台和 AWS CLI 中使用共享子网。

内容

- 共享子网

- 将共享的子网取消共享
- 确定共享子网的拥有者

共享子网

您可以与组织内的其他账户共享非默认子网，如下所示。此外，您还可以跨 AWS Organizations 共享安全组。有关更多信息，请参阅 与 AWS Organizations 共享安全组。

使用控制台共享子网

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Subnets (子网)。
3. 选择您的子网，然后选择操作、共享子网。
4. 选择您的资源共享，然后选择共享子网。

使用 AWS CLI 共享子网

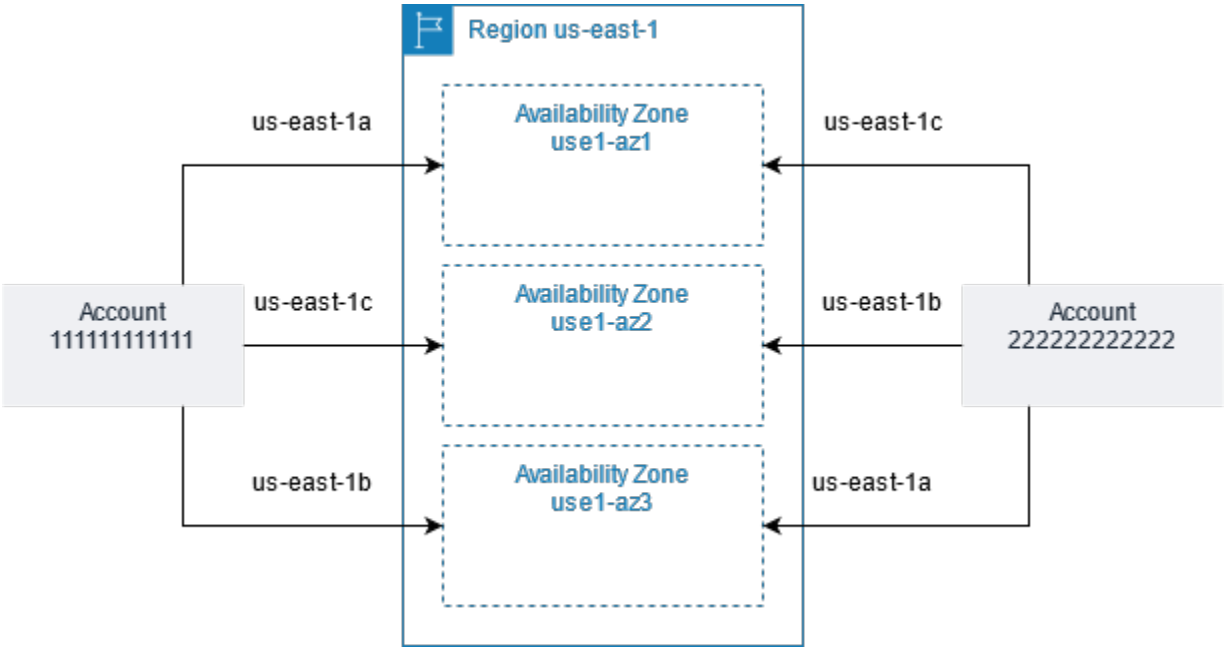
使用 create-resource-share 和 associate-resource-share 命令。

跨可用区映射子网

为确保资源分配到区域的各可用区，我们将可用区独立映射到每个账户的名称。例如，您的 us-east-1a 账户的可用区 AWS 可能与另一 us-east-1a 账户的 AWS 不在同一位置。

要跨账户协调可用区以便进行 VPC 共享，您必须使用 AZ ID（可用区的唯一、一致的标识符）。例如，use1-az1 为 us-east-1 区域中的其中一个可用区的 AZ ID。使用 AZ ID 确定一个账户中的资源相对于另一个账户的位置。您可以在 Amazon VPC 控制台中查看每个子网的 AZ ID。

下图阐明了两个账户，它们具有不同的可用区代码到 AZ ID 的映射。



将共享的子网取消共享

拥有者随时可以将与参与者共享的子网取消共享。在拥有者将共享的子网取消共享后，将应用以下规则：

- 现有参与者资源将继续在已取消共享的子网中运行。具有自动化/托管工作流（如auto 扩展或节点替换）的 AWS 托管服务（例如，Elastic Load Balancing）可能需要持续访问某些资源的共享子网。
- 参与者在已取消共享的子网中无法再创建新资源。
- 参与者可以修改、描述和删除其位于子网中的资源。
- 如果参与者在已取消共享的子网中仍具有资源，则拥有者无法删除共享子网或共享子网 VPC。仅当参与者删除已取消共享的子网中的所有资源之后，拥有者才能删除子网或共享子网 VPC。

使用控制台取消共享子网

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Subnets (子网)。
3. 选择您的子网，然后选择操作、共享子网。
4. 依次选择操作、停止共享。

使用 AWS CLI 取消共享子网

使用 disassociate-resource-share 命令。

确定共享子网的拥有者

参与者可以通过使用 Amazon VPC 控制台或命令行工具来查看已与其共享的子网。

使用控制台确定子网拥有者

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Subnets (子网)。拥有者列显示子网拥有者。

使用 AWS CLI 确定子网拥有者

使用 describe-subnets 和 describe-vpcs 命令，这两条命令的输出中将包括拥有者的 ID。

拥有者与参与者的计费和计量

本节旨在介绍面向共享子网的拥有者与共享子网的使用者的计费和计量详细信息：

- 在共享 VPC 中，每个参与者为其应用程序资源付费，包括 Amazon EC2 实例、Amazon Relational Database Service 数据库、Amazon Redshift 集群和 AWS Lambda 函数。参与者还支付与可用区间数据传输以及跨 VPC 对等连接、跨互联网网关和跨 AWS Direct Connect 网关的数据传输关联的数据传输费用。
- VPC 拥有者支付每小时费用（如果适用），跨 NAT 网关、虚拟私有网关、中转网关、AWS PrivateLink 和 VPC 终端节点的数据处理和数据传输费用。此外，共享 VPC 中使用的公有 IPv4 地址将向 VPC 所有者收费。有关公有 IPv4 地址定价的更多信息，请参阅 Amazon VPC 定价页面中的公有 IPv4 地址定价选项卡。
- 在同一个可用区域（使用 AZ-ID 进行唯一标识）内数据传输是免费的，而不考虑通信资源的账户所有权。

所有者和参与者的责任和权限

本节旨在详细介绍拥有共享子网之人（拥有者）与使用共享子网之人（参与者）的责任和权限。

拥有者资源

拥有者对其拥有的 VPC 资源负责。VPC 拥有者负责创建、管理和删除与共享 VPC 相关的资源。其中包括子网、路由表、网络 ACL、对等连接、网关端点、接口端点、Amazon Route 53 Resolver 端点、互联网网关、NAT 网关、虚拟私有网关和 Transit Gateway 挂载）。

参与者资源

参与者对其拥有的 VPC 资源负责。参与者可以在共享 VPC 中创建一组有限的 VPC 资源。例如，参与者可以创建网络接口和安全组，此外还可为其拥有的网络接口启用 VPC 流日志。参与者创建的 VPC 资源将计入参与者账户中的 VPC 限额，而不是所有者账户的限额。有关更多信息，请参阅 VPC 子网共享。

VPC 资源

使用共享 VPC 子网时，以下责任和权限适用于 VPC 资源：

流日志

- 参与者可以在共享 VPC 子网中为他们拥有的网络接口创建、删除和描述流日志。
- 参与者不能在共享 VPC 子网中为他们不拥有的网络接口创建、删除或描述流日志。
- 参与者不能为共享 VPC 子网创建、删除或描述流日志。
- VPC 所有者能在共享 VPC 子网中为他们不拥有的网络接口创建、删除和描述流日志。
- VPC 所有者能为共享 VPC 子网创建、删除和描述流日志。
- VPC 所有者无法描述或删除参与者创建的流日志。

互联网网关和仅出口互联网网关

- 参与者无法在共享 VPC 子网中创建、附加或删除互联网网关和仅出口互联网网关。参与者可以在共享 VPC 子网中描述互联网网关。参与者无法在共享 VPC 子网中描述仅出口互联网网关。

NAT 网关

- 参与者无法在共享 VPC 子网中创建、删除或描述 NAT 网关。

网络访问控制列表（NACL）

- 参与者无法在共享 VPC 子网中创建、删除或替换 NACL。参与者可以在共享 VPC 子网中描述 VPC 所有者创建的 NACL。

网络接口

- 参与者可以在共享 VPC 子网中创建网络接口。参与者无法以任何其他方式（例如附加、分离或修改网络接口）在共享 VPC 子网中使用 VPC 所有者创建的网络接口。参与者可以在他们创建的共享 VPC 中修改或删除网络接口。例如，参与者可以将 IP 地址与他们创建的网络接口关联或解除关联。
- VPC 所有者可以在共享 VPC 子网中描述参与者拥有的网络接口。VPC 所有者无法以任何其他方式（例如附加、分离或修改网络接口）在共享 VPC 子网中使用参与者拥有的网络接口。

路由表

- 参与者无法在共享 VPC 子网中使用路由表（例如，创建、删除或关联路由表）。参与者可以在共享 VPC 子网中描述路由表。

安全组

- 参与者可以在共享 VPC 子网中使用其拥有的安全组（创建、删除、描述、修改或创建传入和传出规则）。如果 VPC 所有者与参与者共享安全组，则参与者可以使用 VPC 所有者创建的安全组。
- 参与者可以在其拥有的安全组中创建规则，并引用属于其他参与者或 VPC 所有者的安全组，如下所示：account-number/security-group-id
- 参与者无法使用 VPC 的默认安全组启动实例，因为此安全组属于所有者。
- 参与者无法使用 VPC 所有者或其他参与者拥有的非默认安全组启动实例，除非与他们共享安全组。
- VPC 所有者可以在共享 VPC 子网中描述参与者创建的安全组。VPC 所有者无法以任何其他方式使用参与者创建的安全组。例如，VPC 所有者无法使用参与者创建的安全组启动实例。

子网

- 参与者无法修改共享子网或这些子网的相关属性。只有 VPC 所有者可以。参与者可以在共享 VPC 子网中描述子网。
- VPC 所有者只能通过 AWS Organizations 与同一组织的其他账户或组织单位共享子网。VPC 所有者无法共享位于默认 VPC 中的子网。

中转网关

- 只有 VPC 所有者可以将中转网关附加到共享 VPC 子网。参与者不能。

VPC

- 参与者无法修改 VPC 或 VPC 的相关属性。只有 VPC 所有者可以。参与者可以描述 VPC、VPC 属性和 DHCP 选项集。
- VPC 标签和共享 VPC 内资源的标签不会与参与者共享。
- 参与者可以将自己的安全组与共享 VPC 关联。这样参与者便可将安全组与其在共享 VPC 中拥有的弹性网络接口一起使用。

AWS 资源和共享 VPC 子网

以下 AWS 服务 支持共享 VPC 子网中的资源。有关更多信息，请访问相应服务文档的链接。

- Amazon Aurora
- AWS Database Migration Service
- Amazon EC2
- Amazon ECS
- Amazon ElastiCache（Redis OSS）
- Amazon EFS
- Amazon Elastic Kubernetes Service
- Elastic Load Balancing
  - 应用程序负载均衡器
  - 网关负载均衡器
  - 网络负载均衡器
- Amazon EMR
- AWS Glue
- AWS Lambda
- 运行 Apache MQ（而非 Rabbit MQ）的 Amazon MQ
- Amazon MSK
- AWS Network Manager
  - AWS Cloud WAN
  - 网络访问分析器
  - Reachability Analyzer

- Amazon OpenSearch Service
- AWS PrivateLink†
- Amazon Relational Database Service (RDS)
- Amazon Redshift
- Amazon Route 53
- AWS Transit Gateway
- AWS Verified Access
- Amazon VPC
  - 对等连接
  - 流量镜像
- Amazon VPC Lattice

† 您可以使用共享 VPC 中的 VPC 端点连接到支持 PrivateLink 的所有 AWS 服务。有关支持 PrivateLink 的服务列表，请参阅《AWS PrivateLink 指南》中的与 AWS PrivateLink 集成的 AWS 服务。

此列表旨在列出所有支持在共享 VPC 子网中启动资源的服务。尽管我们已尽最大努力，但仍可能还有其他未在此列出的服务支持在共享 VPC 子网中启动资源。如有疑问，建议您提交文档反馈。

将 VPC 扩展到本地区域、Wavelength 区域或 Outpost

您可以在全球多个位置托管 VPC 资源（如子网）。这些位置由“区域”、“可用区”、“本地区域”和“Wavelength 区域”组成。每个区域都是一个单独的地理区域。

- 可用区是每个区域内的多个相互隔离的位置。
- 本地区域允许您在多个离终端用户较近的位置放置资源（如计算和存储）。
- AWS Outposts 可将本机 AWS 服务、基础设施和运营模式引入几乎任何数据中心、主机托管空间或本地设施。
- 利用 Wavelength 区域，开发人员可以为 5G 设备和最终用户打造具有超低延迟的应用程序。Wavelength 可以将标准 AWS 计算和存储服务部署到电信运营商的 5G 网络边缘。

AWS运行着具有高可用性的先进数据中心。数据中心有时会发生影响托管于同一位置的所有实例的可用性的故障，虽然这种故障极少发生。如果您将所有实例都托管在受故障影响的同一个位置，则您的所有实例都将不可用。

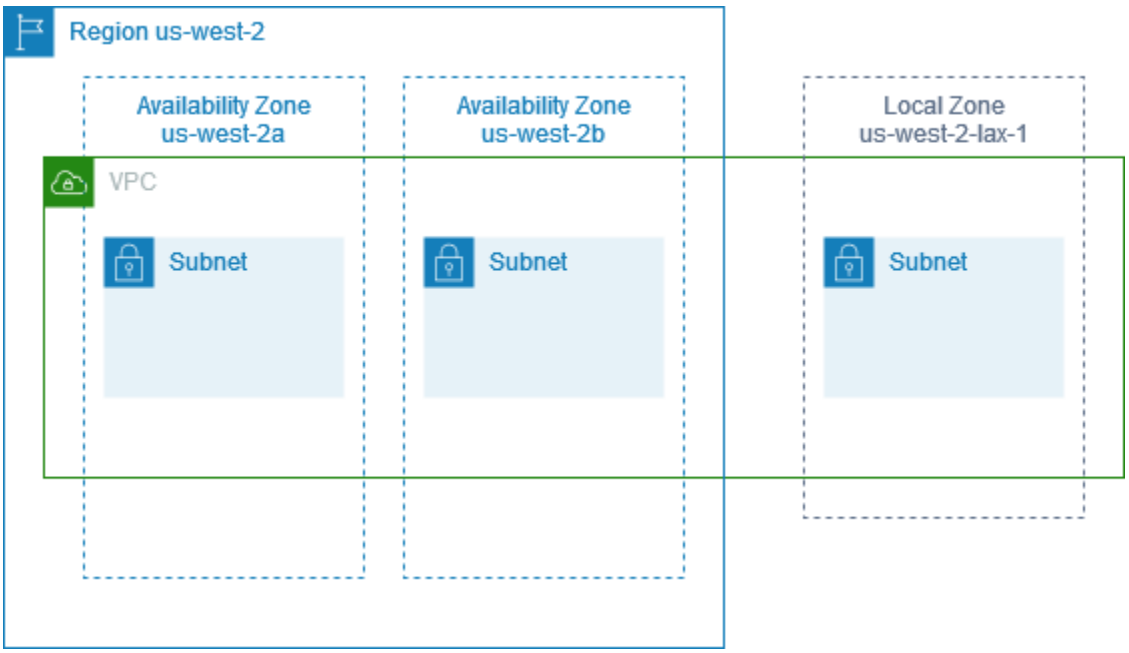
AWS Local Zones 中的子网

AWS Local Zones 允许您在靠近用户的位置放置资源，并且使用熟悉的 API 和工具集无缝连接到 AWS 区域中的各种服务。当您在本地区域中创建子网时，您也会将 VPC 扩展到该本地区域。

要使用 Local Zone，您需要遵循以下流程：

- 选择加入 Local Zone。
- 在本地扩展区中创建子网。
- 在 Local Zone 子网中启动资源，以确保您的应用程序靠近用户。

下图演示的 VPC 位于美国西部（俄勒冈州）(us-west-2) 区域，横跨多个可用区和一个 Local Zone。



创建 VPC 时，您可以选择为 VPC 分配一组由 Amazon 提供的公有 IP 地址。您还可以为这些地址设置网络边界组，以将地址限制到该组。设置网络边界组时，IP 地址不能在网络边界组间移动。本地区域网络流量将直接进入互联网或接入网点 (PoP)，无需遍历本地区域的父区域，从而能够访问低延迟计算。对于 Local Zones 及其相应父区域的完整列表，请参阅 AWS Local Zones 用户指南中的可用的 Local Zones。

以下规则适用于本地区域：

- 本地区域子网遵循与可用区子网相同的路由规则，包括路由表、安全组和网络 ACL。
- 出站互联网流量从本地区域内部离开。

- 您必须预配置公有 IP 地址以便在本地区域中使用。分配地址时，您可以指定通告其中 IP 地址的位置。我们将其称为网络边界组，您可以设置此参数，将地址限制到此位置。预置 IP 地址后，您无法在本地区域与父区域之间移动这些地址（例如，从 `us-west-2-lax-1a` 到 `us-west-2`）。
- 如果 Local Zone 支持 IPv6，您可以请求 Amazon 提供的 IPv6 IP 地址，并将其与新 VPC 或现有 VPC 的网络边界组关联。有关支持 IPv6 的 Local Zones 列表，请参阅 AWS Local Zones 用户指南中的注意事项
- 您无法在 Local Zone 子网内创建 VPC 端点。

有关使用 Local Zones 的更多信息，请参阅 AWS Local Zones 用户指南。

互联网网关的注意事项

在本地区域中使用（父区域中的）互联网网关时，请考虑以下信息：

- 您可以在本地区域中使用具有弹性 IP 地址或 Amazon 自动分配的公有 IP 地址的互联网网关。您关联的弹性 IP 地址必须包括本地区域的网络边界组。有关更多信息，请参阅 the section called “弹性 IP 地址”。

  您不能关联为该区域设置的弹性 IP 地址。
- 本地区域中使用的弹性 IP 地址与区域中的弹性 IP 地址在配额上相同。有关更多信息，请参阅 the section called “弹性 IP 地址”。
- 您可以在与本地区域资源关联的路由表中使用互联网网关。有关更多信息，请参阅 the section called “路由到互联网网关”。

使用 Direct Connect 网关访问本地区域

考虑一下您希望本地数据中心访问本地扩展区中的资源的情况。您可以将虚拟私有网关用于与本地区域关联的 VPC，以连接到 Direct Connect 网关。Direct Connect 网关连接到区域中的 AWS Direct Connect 站点。本地部署数据中心拥有与该 AWS Direct Connect 位置的 AWS Direct Connect 连接。

> **Note**
>
> 使用 Direct Connect 发往本地区域子网的流量不会通过本地区域的父区域。相反，流量会采用最短路径到达本地区域。这可以减少延迟，并有助于提高应用程序的响应速度。

对于此配置，可以配置以下资源：

- 与本地扩展区子网关联的 VPC 的虚拟私有网关。您可以在 Amazon Virtual Private Cloud Console 中的子网详细信息页面上查看子网的 VPC，也可以使用 describe-subnets 命令。

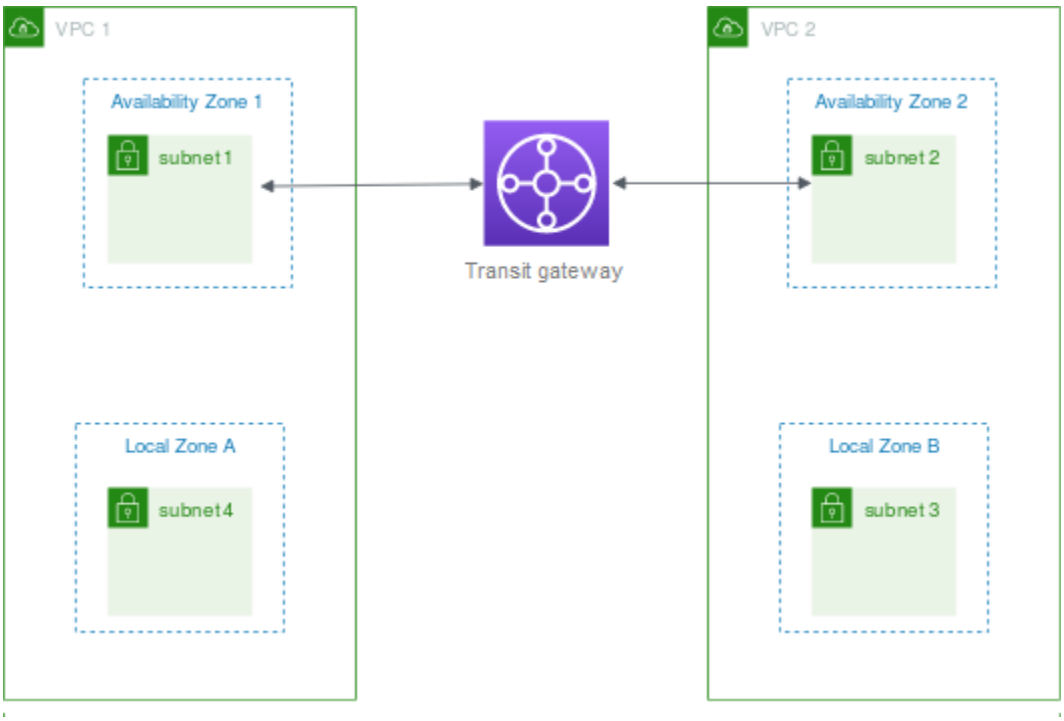
  有关如何创建虚拟私有网关的信息，请参阅《AWS Site-to-Site VPN 用户指南》中的创建目标网关。
- Direct Connect 连接。为了获得最佳延迟性能，AWS 建议您使用要将子网扩展到的最靠近本地区域的 Direct Connect 站点。

  有关如何订购连接的信息，请参阅《AWS Direct Connect 用户指南》中的交叉连接。
- 一个 Direct Connect 网关。有关如何创建 Direct Connect 网关的信息，请参阅《AWS Direct Connect 用户指南》中的创建 Direct Connect 网关。
- 将 VPC 连接到 Direct Connect 网关的虚拟私有网关关联。有关如何创建虚拟私有网关关联的信息，请参阅《AWS Direct Connect 用户指南》中的关联和取消关联虚拟私有网关。
- 从 AWS Direct Connect 站点到本地部署数据中心的连接上的私有虚拟接口。有关如何创建 Direct Connect 网关的信息，请参阅《AWS Direct Connect 用户指南》中的创建到 Direct Connect 网关的私有虚拟接口。

将本地扩展区子网连接到中转网关

您无法为本地区域中的子网创建中转网关连接。下图显示了如何配置网络，以便本地区域中的子网通过父可用区连接到中转网关。在本地区域中创建子网，并在父可用区中创建子网。将父可用区域中的子网连接到中转网关，然后在路由表中为每个 VPC 创建一个路由，该路由将用于其他 VPC CIDR 的流量路由到中转网关连接的网络接口。

> **Note**
>
> 从中转网关发往本地区域中子网的流量将首先遍历父区域。



为此场景创建以下资源：

- 每个父可用区中的子网。有关更多信息，请参阅 the section called “创建子网”。
- 中转网关。有关更多信息，请参阅 Amazon VPC Transit Gateway 中的创建中转网关。
- 使用父可用区的每个 VPC 的中转网关连接。有关更多信息，请参阅《Amazon VPC Transit Gateway》中的 Create a transit gateway attachment to a VPC。
- 与中转网关连接关联的中转网关路由表。有关更多信息，请参阅 Amazon VPC Transit Gateway中的中转网关路由表。
- 对于每个 VPC，在本地区域子网的子网路由表中创建一个条目，该条目需将另一个 VPC CIDR 作为目的地，并将中转网关连接的网络接口 ID 作为目标。要查找中转网关连接的网络接口，请在网络接口的说明中搜索中转网关连接的 ID。有关更多信息，请参阅 the section called “中转网关的路由”。

以下是 VPC 1 的示例路由表。

目标位置	目标
VPC 1 CIDR	*##*
VPC 2 CIDR	

目标位置	目标
	vpc1-attachment-network-interface-id

以下是 VPC 2 的示例路由表。

目标位置	目标
VPC 2 CIDR	*##*
VPC 1 CIDR	*vpc2-attachment-network-interface-id*

以下是中转网关路由表的示例。每个 VPC 的 CIDR 块将传播到中转网关路由表。

CIDR	附件	路由类型
VPC 1 CIDR	*VPC 1 ###*	传播
VPC 2 CIDR	*VPC 2 ###*	传播

AWS Wavelength 中的子网

利用 AWS Wavelength，开发人员可以为移动设备和最终用户打造具有超低延迟的应用程序。Wavelength 可以将标准 AWS 计算和存储服务部署到电信运营商的 5G 网络边缘。开发人员可以将虚拟私有云（VPC）扩展到一个或多个 Wavelength Zone，然后使用 Amazon EC2 实例等 AWS 资源来运行需要超低延迟并连接到区域中的 AWS 服务的应用程序。

要使用 Wavelength 区域，必须首先选择加入区域。接下来，在 Wavelength 区域中创建子网。您可以在 Wavelength 区域中创建 Amazon EC2 实例、Amazon EBS 卷和 Amazon VPC 子集和 Carrier Gateway。此外，您还可以使用通过 EC2、EBS 和 VPC 编排或搭配使用的服务，如 Amazon EC2 Auto Scaling、Amazon EKS 集群、Amazon ECS 集群、Amazon EC2 Systems Manager、Amazon CloudWatch、AWS CloudTrail 和 AWS CloudFormation。Wavelength 中的服务是 VPC 的一部分，它

通过可靠的高带宽连接至 AWS 区域，以便轻松访问包括 Amazon DynamoDB 和 Amazon RDS 在内的服务。

以下规则适用于 Wavelength 区域：

- 当您在 VPC 中创建子网并将其与 Wavelength 区域关联时，VPC 将扩展到 Wavelength 区域。
- 默认情况下，您在跨越某个 Wavelength 区域的 VPC 中创建的每个子网都会继承主 VPC 路由表，包括本地路由。
- 当您在 Wavelength 区域的子网中启动 EC2 实例时，您将为其分配运营商 IP 地址。运营商网关将地址用于从接口到 Internet 或移动设备的流量。运营商网关使用 NAT 转换地址，然后将流量发送到目的地。来自电信运营商网络的流量通过运营商网关路由。
- 您可以将 VPC 路由表或 Wavelength 区域中的子网路由表的目标设置为运营商网关，从而允许来自特定位置的运营商网络的入站流量，以及向运营商网络和 Internet 发送出站流量。有关 Wavelength 区域中的路由选项的更多信息，请参阅《AWS Wavelength 开发人员指南》中的路由。
- Wavelength 区域中的子网与可用区中的子网具有相同的网络组件，包括 IPv4 地址、DHCP 选项集和网络 ACL。
- 您无法为 Wavelength 区域中的子网创建中转网关连接。但可以通过父可用区中的子网创建附件，然后通过 Transit Gateway 将流量路由到所需目的地。有关示例，请参阅下一节。

存在多个 Wavelength 区域时的注意事项

不允许位于同一 VPC 中不同 Wavelength 区域内的 EC2 实例之间相互通信。如果您需要在不同 Wavelength 区域之间进行通信，AWS 建议您使用多个 VPC，每个 Wavelength 区域一个。您可以使用中转网关连接这些 VPC。此配置允许这些 Wavelength 区域中的实例之间相互通信。

Wavelength 区域到 Wavelength 区域的流量传输会经过 AWS 区域。有关更多信息，请参阅 AWS Transit Gateway。

下图显示了如何配置网络以支持两个不同 Wavelength 区域中的实例相互通信。您有两个 Wavelength 区域（Wavelength 区域 A 和 Wavelength 区域 B）。您需要创建以下资源才能支持通信：

- 对于每个 Wavelength 区域，创建一个位于其父可用区中的子网。在示例中，您创建了子网 1 和子网 2。有关创建子网的信息，请参阅 the section called “创建子网”。使用 describe-availability-zones 命令查找父区域。
- 中转网关。中转网关连接 VPC。有关如何创建中转网关的信息，请参阅《Amazon VPC Transit Gateway 指南》中的创建中转网关。

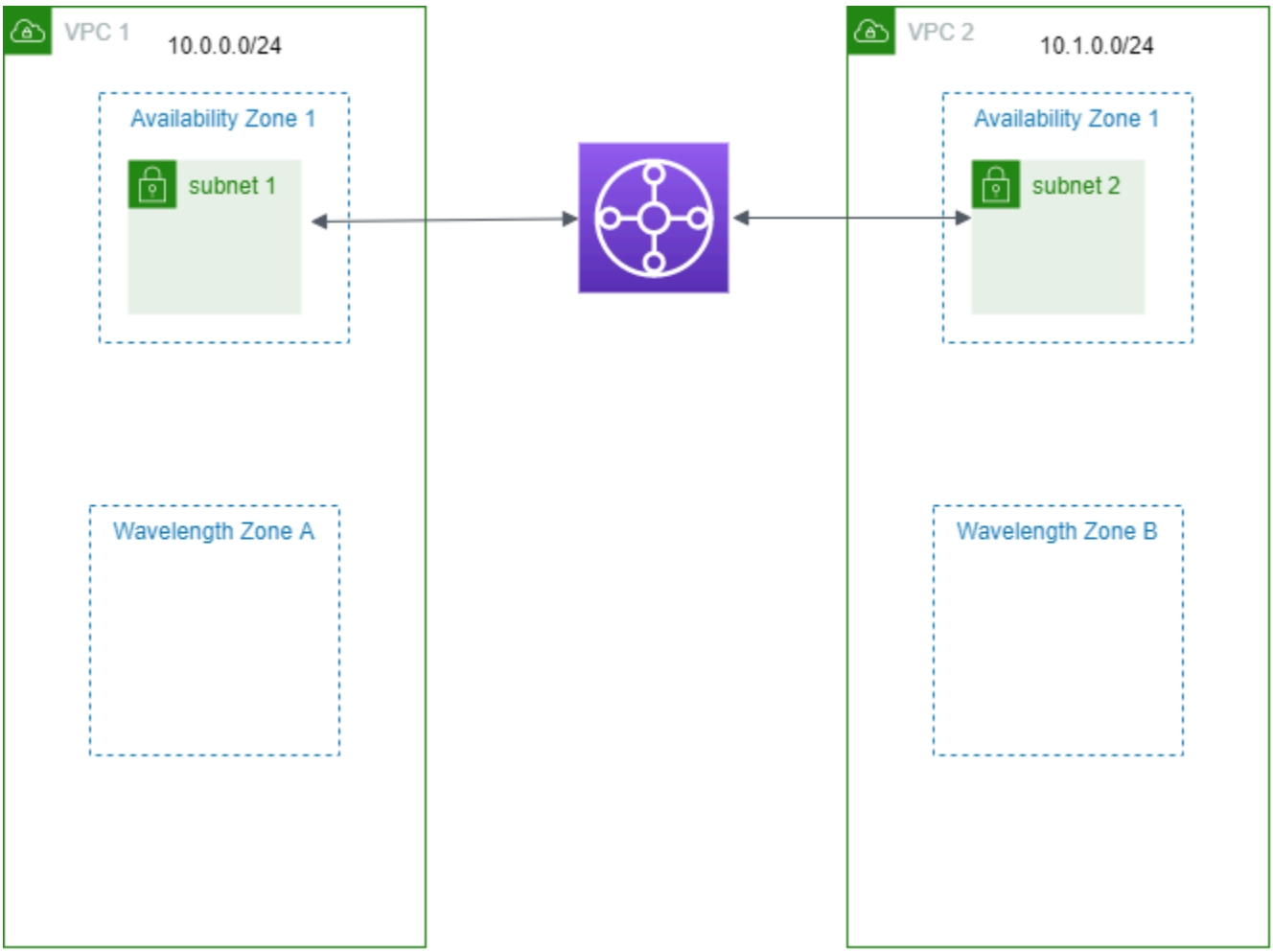
- 对于每个 VPC，VPC 挂载到 Wavelength 区域父可用区的 Transit Gateway 中。有关更多信息，请参阅《Amazon VPC Transit Gateway 指南》中的 Transit gateway attachments to a VPC。
- 中转网关路由表中每个 VPC 的条目。有关如何创建 Transit Gateway 路由的信息，请参阅《Amazon VPC Transit Gateway 指南》中的中转网关路由表。
- 对于每个 VPC，在 VPC 路由表中创建一个条目，该条目需将另一个 VPC CIDR 作为目的地，并将中转网关 ID 作为目标。有关更多信息，请参阅 the section called “中转网关的路由”。

在示例中，VPC 1 的路由表包含以下条目：

目的地	目标
10.1.0.0/24	tgw-22222222222222222

VPC 2 的路由表包含以下条目：

目的地	目标
10.0.0.0/24	tgw-22222222222222222



AWS Outposts 中的子网

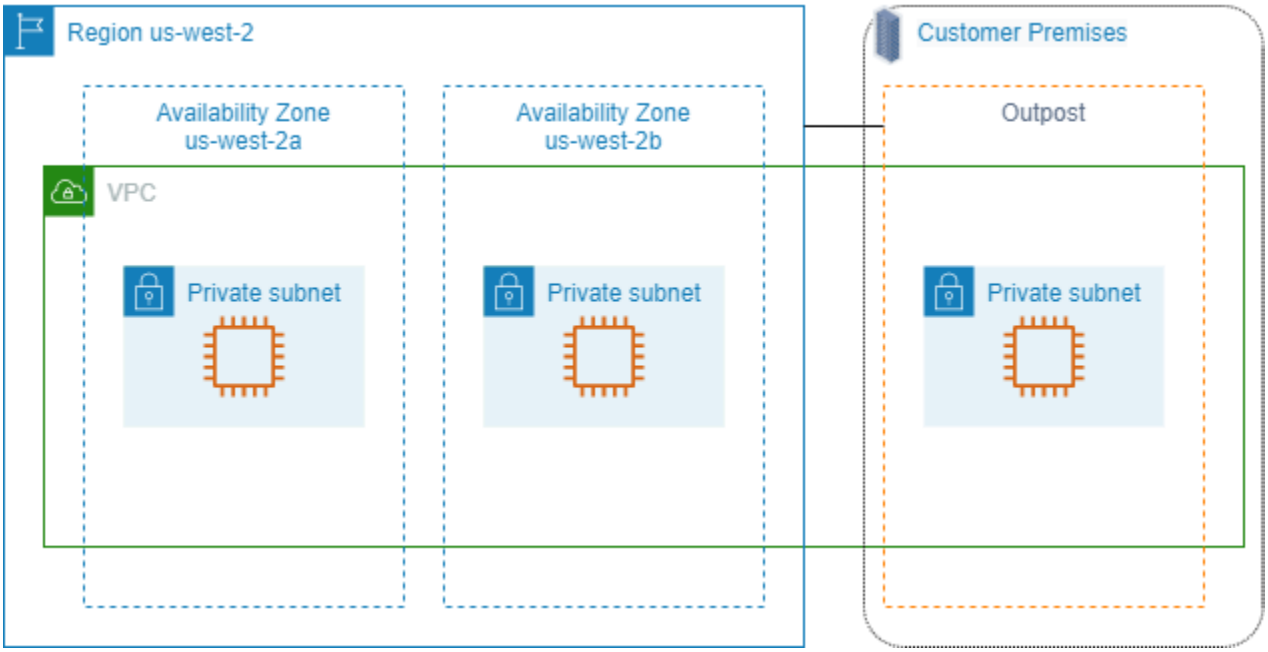
AWS Outposts 为您提供相同的 AWS 硬件基础设施、服务、API 和工具，用于在本地和云中构建并运行您的应用程序。AWS Outposts 非常适合需要以低延迟方式访问本地应用程序或系统的工作负载，以及需要在本地存储和处理数据的工作负载。有关 AWS Outposts 的更多信息，请参阅 AWS Outposts。

VPC 涵盖一个 AWS 区域中的所有可用区。将您的 Outpost 连接到其父级区域后，您可以在该 VPC 中为您的 Outpost 创建子网，从而将该区域中的所有 VPC 都扩展到您的 Outpost。

以下规则适用于 AWS Outposts：

- 子网必须位于一个 Outposts 位置。
- 要为 Outpost 创建子网，请在创建子网时指定 Outpost 的Amazon 资源名称（ARN）。
- Outposts 机架 – 由本地网关处理 VPC 与本地网络之间的网络连接。有关更多信息，请参阅《AWS Outposts 用户指南》中的 本地网关。

- Outposts 服务器 – 由本地网络接口处理 VPC 与本地网络之间的网络连接。有关更多信息，请参阅《适用于 Outposts 服务器的 AWS Outposts 用户指南》中的 本地网络接口。
- 默认情况下，您在 VPC 中创建的每个子网（包括为您的 Outpost 创建的子网），都会隐式与 VPC 的主路由表关联。您还可以将自定义路由表与 VPC 中的子网显式关联，并将本地网关作为指向本地网络的所有流量的下一跳目标。



删除您的 VPC

用完 VPC 后可以将其删除。

要求

在删除 VPC 之前，必须先终止或删除在 VPC 中创建了请求者托管式网络接口的任何资源。例如，您必须终止 EC2 实例并删除负载均衡器、NAT 网关、中转网关 VPC 挂载和接口 VPC 端点。

> **Note**
>
> 如果您已为要删除的 VPC 创建了流日志，请注意，已删除的 VPC 的流日志最终会自动删除。

内容

- 使用控制台删除 VPC

- 使用命令行删除 VPC

使用控制台删除 VPC

如果您使用 Amazon VPC 控制台删除 VPC，我们还会为您删除以下 VPC 组件：

- DHCP 选项
- 仅出口互联网网关
- 网关端点
- Internet 网关
- 网络 ACL
- 路由表
- 安全组
- 子网

使用控制台删除 VPC

1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 终止 VPC 中的所有实例。有关更多信息，请参阅《Amazon EC2 用户指南》中的终止实例。
3. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
4. 在导航窗格中，选择 Your VPCs（您的 VPC）。
5. 选择要删除的 VPC，然后依次选择 Actions（操作）、Delete VPC（删除 VPC）。
6. 如果存在您必须首先删除或终止的资源，然后我们才能删除 VPC，则我们会显示这些资源。请删除或终止这些资源，然后重试。否则，我们将显示除 VPC 之外还要删除的资源。检查列表，然后继续执行下一步操作。
7. （可选）如果您有 Site-to-Site VPN 连接，则可以选择此选项以将其删除。如果您计划在另一个 VPC 中使用客户网关，我们建议您保留 Site-to-Site VPN 连接和网关。否则，您必须在创建新的 Site-to-Site VPN 连接后再次配置客户网关设备。
8. 提示进行确认时，输入 **`delete`**，然后选择 Delete（删除）。

使用命令行删除 VPC

在使用命令行删除 VPC 之前，必须先终止或删除在 VPC 中创建了请求者托管式网络接口的任何资源。您还必须删除或分离您创建的所有资源，例如子网、安全组、网络 ACL、路由表、互联网网关和仅限出口的互联网网关。您无需删除默认安全组、默认路由表或默认网络 ACL。

以下过程演示了用于删除常见 VPC 资源，然后删除 VPC 的命令。您必须按照以下顺序使用这些命令。如果您创建了其他 VPC 资源，则还需要使用其相应的删除命令，然后才能删除 VPC。

使用 AWS CLI 删除 VPC

1. 使用 delete-security-group 命令删除安全组。

```
aws ec2 delete-security-group --group-id sg-id
```

2. 使用 delete-network-acl 命令删除每个网络 ACL。

```
aws ec2 delete-network-acl --network-acl-id acl-id
```

3. 使用 delete-subnet 命令删除每个子网。

```
aws ec2 delete-subnet --subnet-id subnet-id
```

4. 使用 delete-route-table 命令删除每个自定义路由表。

```
aws ec2 delete-route-table --route-table-id rtb-id
```

5. 使用 detach-internet-gateway 命令将互联网网关与 VPC 分离。

```
aws ec2 detach-internet-gateway --internet-gateway-id igw-id --vpc-id vpc-id
```

6. 使用 delete-internet-gateway 命令删除互联网网关。

```
aws ec2 delete-internet-gateway --internet-gateway-id igw-id
```

7. [双堆栈 VPC] 使用 delete-egress-only-internet-gateway 命令删除仅限出口的互联网网关。

```
aws ec2 delete-egress-only-internet-gateway --egress-only-internet-gateway-id eigw-id
```

8. 使用 delete-vpc 命令删除 VPC。

```
aws ec2 delete-vpc --vpc-id vpc-id
```

使用 Console-to-Code，通过您的 VPC 控制台操作生成基础设施即代码

控制台提供了一条用来创建资源和测试原型的引导式路径。如果希望大规模创建相同的资源，则您需要使用自动化代码。Console-to-Code 是 Amazon Q 开发者版的一项功能，可以帮助您开始使用自动化代码。Console-to-Code 会记录您的控制台操作，包括默认值和兼容参数。随后，其利用生成式人工智能，以您首选的基础设施即代码（IaC）格式，为您要执行的操作提供代码建议。由于控制台工作流程可确保您指定的参数值同时有效，因此，您使用 Console-to-Code 生成的代码具有兼容的参数值。您可以将此代码用作一个起点，然后对其进行自定义，以使它可用于您的特定使用案例的生产。

例如，通过 Console-to-Code，您可以使用 VPC 控制台记录自己创建子网、安全组、NACL、自定义路由表和互联网网关，并生成 AWS CloudFormation JSON 格式的代码。然后，您可以复制该代码并对其进行自定义，以便在 AWS CloudFormation 模板中使用。

Console-to-Code 目前能够以下列语言和格式生成基础设施即代码（IaC）：

- CDK Java
- CDK Python
- CDK TypeScript
- CloudFormation JSON
- CloudFormation YAML

有关如何使用 Console-to-Code 的更多信息和说明，请参阅《Amazon Q Developer User Guide》中的 Automating AWS services with Amazon Q Developer Console-to-Code。

VPC 的子网

子网是您的 VPC 内的 IP 地址范围。您可以在特定子网中创建 AWS 资源（例如 EC2 实例）。

内容

- 子网基础知识
- 子网安全性
- 创建子网
- 将 IPv6 CIDR 块添加到子网或从中删除
- 修改子网的 IP 寻址属性
- 子网 CIDR 预留
- 配置路由表
- 中间盒路由向导
- 删除子网

子网基础知识

每个子网都必须完全位于一个可用区之内，不能跨越多个可用区。通过在独立的可用区内启动 AWS 资源，可以保护应用程序不受单一可用区故障的影响。

内容

- 子网 IP 地址范围
- 子网类型
- 子网图
- 子网路由
- 子网设置

子网 IP 地址范围

在创建子网时，您可以根据 VPC 的配置指定其 IP 地址：

- 仅 IPv4 – 子网具有 IPv4 CIDR 块，但没有 IPv6 CIDR 块。仅限 IPv4 的子网中的资源必须通过 IPv4 进行通信。

- 双堆栈 – 子网同时具有 IPv4 CIDR 块和 IPv6 CIDR 块。VPC 必须同时具有 IPv4 CIDR 块和 IPv6 CIDR 块。双堆栈子网中的资源可以通过 IPv4 和 IPv6 进行通信。
- 仅 IPv6 – 子网具有 IPv6 CIDR 块，但没有 IPv4 CIDR 块。VPC 必须具有一个 IPv6 CIDR 块。仅限 IPv6 的子网中的资源必须通过 IPv6 进行通信。

> **Note**
>
> 仅限 IPv6 的子网中的资源从 CIDR 块 169.254.0.0/16 中分配了 IPv4 链路本地地址。这些地址用于与仅在 VPC 中可用的服务进行通信。有关示例，请参阅《Amazon EC2 用户指南》中的链路本地地址。

有关更多信息，请参阅 为 VPC 和子网分配 IP 地址。

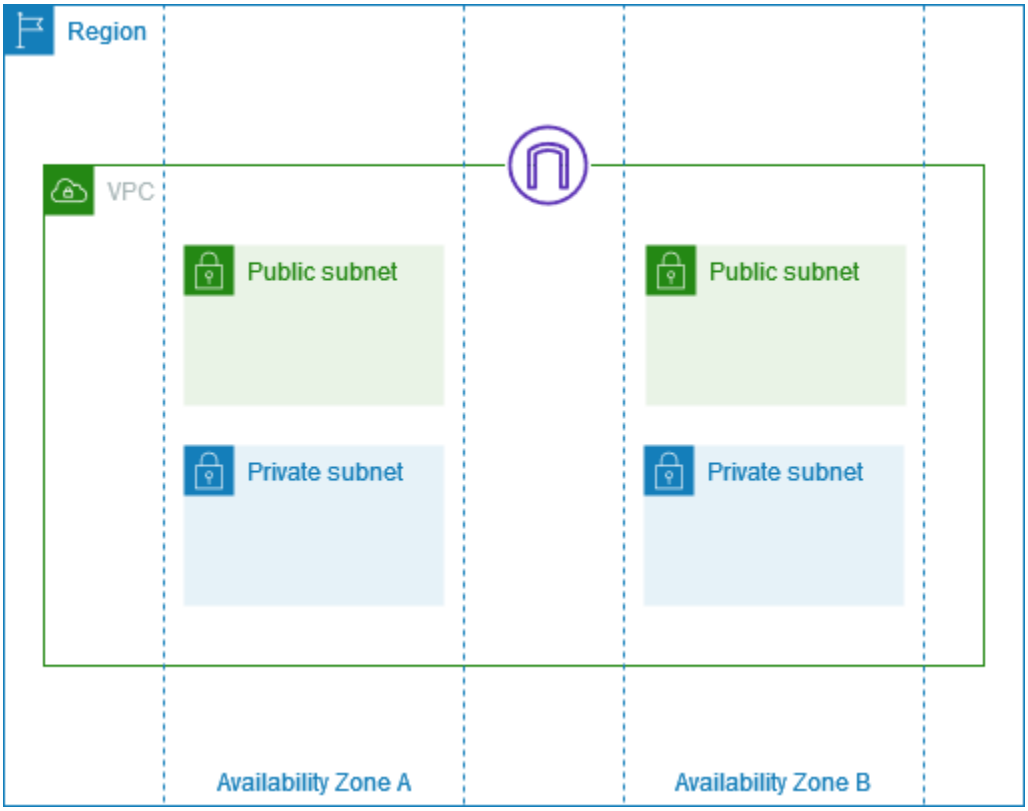
子网类型

子网类型取决于您如何为子网配置路由。例如：

- 公有子网 – 子网具有一条指向某个 互联网网关 的直接路由。公有子网中的资源可以访问公有互联网。
- 私有子网 – 子网不具有指向任何互联网网关的直接路由。私有子网中的资源需要使用 NAT 设备才能访问公有互联网。
- 仅限 VPN 的子网 – 子网具有一个通过虚拟私有网关指向某个 Site-to-Site VPN 连接 的路由。该子网不具有通向互联网网关的路由。
- 隔离子网 – 子网没有通往其 VPC 之外目的地的路由。隔离子网中的资源只能访问同一 VPC 中的其他资源或被同一 VPC 中的其他资源访问。
- EVS 子网 – 此类子网是使用 Amazon EVS 创建的。有关更多信息，请参阅《Amazon EVS 用户指南》中的 VLAN subnet。

子网图

下图显示了具有两个可用区中的子网和一个互联网网关的 VPC。每个可用区都有一个公有子网和一个私有子网。



有关显示本地区域和 Wavelength 区域中子网的示意图，请参阅 How AWS Local Zones work 和 How AWS Wavelength works。

子网路由

每个子网都必须关联一个路由表，这个路由表可指定允许出站流量离开子网的可用路由。您创建的每个子网都会自动关联 VPC 的主路由表。您可以更改关联，以及更改主路由表的内容。有关更多信息，请参阅 配置路由表。

子网设置

所有子网都有一个用于确定是否向在该子网中创建的网络接口分配公有 IPv4 地址和 IPv6 地址 (如果适用) 的可修改属性。这包括当您在该子网中启动实例时为实例创建的主网络接口（例如，eth0）。不管子网属性如何，您仍然可以在启动时覆盖特定实例的此设置。

创建子网后，您可以修改子网的以下设置：

- 自动分配 IP 设置：允许您配置自动分配 IP 设置，以便为此子网中的新网络接口自动请求公有 IPv4 或 IPv6 地址。

- 基于资源的名称 (RBN) 设置：允许您为此子网中的 EC2 实例指定主机名类型，并配置 DNS A 和 AAAA 记录查询的处理方式。有关更多信息，请参阅《Amazon EC2 用户指南》中的 Amazon EC2 实例主机名类型。

子网安全性

为了保护您的 AWS 资源，我们建议您使用私有子网。使用堡垒主机或 NAT 设备为私有子网中的资源提供互联网访问，例如 EC2 实例。

AWS 提供了多种可以用于提高 VPC 中资源安全性的功能。安全组用于允许关联资源（例如 EC2 实例）的入站和出站流量。网络 ACL 用于允许或拒绝子网级别的入站和出站流量。在大多数情况下，使用安全组即可满足您的需求。不过，如需为 VPC 增加额外的安全保护，您可以使用网络 ACL。有关更多信息，请参阅 the section called “比较安全组和网络 ACL”。

每个子网有意必须与一个网络 ACL 关联。您创建的每个子网均自动与 VPC 的原定设置网络 ACL 关联。默认网络 ACL 允许所有入站和出站流量。您可以更新默认网络 ACL，也可以创建自定义网络 ACL 并将其与您的子网关联。有关更多信息，请参阅 使用网络访问控制列表控制子网流量。

您可以在 VPC 或子网上创建流日志，以便捕获传入和传出您的 VPC 或子网中的网络接口的流量。您还可以在单独的网络接口上创建流日志。有关更多信息，请参阅 使用 VPC 流日志记录 IP 流量。

创建子网

按照以下过程为您的虚拟私有云（VPC）创建子网。根据需要的连接，您可能还需要添加网关和路由表。

注意事项

- 您必须在 VPC 范围内为子网指定 IPv4 CIDR 块。如果 VPC 已关联了 IPv6 CIDR 块，则可以选择为子网指定 IPv6 CIDR 块。有关更多信息，请参阅 为 VPC 和子网分配 IP 地址。
- 如果创建仅使用 IPv6 的子网，请注意以下事项。在仅使用 IPv6 的子网中启动的 EC2 实例将会获得 IPv6 地址，但不会获得 IPv4 地址。您在仅使用 IPv6 的子网中启动的任何实例，必须是基于 Nitro 系统构建的实例。
- 要在本地区域或 Wavelength 区域中创建子网，必须启用该区域。有关更多信息，请参阅《Amazon EC2 用户指南》中的区域和区。

为您的 VPC 添加子网

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Subnets(子网)。
3. 选择创建子网。
4. 在 VPC ID 下选择该子网的 VPC。
5. （可选）对于 Subnet name（子网名称），输入子网的名称。这样做可创建具有 Name 键以及您指定的值的标签。
6. 对于 Availability Zone（可用区），您可以为子网选择一个可用区，也可保留原定设置 No Preference（无首选项），以让 AWS 代您选择。
7. 对于 IPv4 CIDR 块，选择手动输入后输入子网的 IPv4 CIDR 块（例如，`10.0.1.0/24`），或者选择无 IPv4 CIDR。如果您使用 Amazon VPC IP 地址管理器（IPAM）规划、跟踪和监控 AWS 工作负载的 IP 地址，则在创建子网时，您可以选择从 IPAM 分配 CIDR 块（IPAM-allocated）。有关为子网 IP 分配规划 VPC IP 地址空间的更多信息，请参阅《Amazon VPC IPAM 用户指南》中的教程：为子网 IP 分配规划 VPC IP 地址空间。
8. 对于 IPv6 CIDR 块，选择手动输入以选择要在其中创建子网的 VPC 的 IPv6 CIDR。此选项仅在 VPC 已经关联了 IPv6 CIDR 块时可用。如果您使用 Amazon VPC IP 地址管理器（IPAM）规划、跟踪和监控 AWS 工作负载的 IP 地址，则在创建子网时，您可以选择从 IPAM 分配 CIDR 块（IPAM-allocated）。有关为子网 IP 分配规划 VPC IP 地址空间的更多信息，请参阅《Amazon VPC IPAM 用户指南》中的教程：为子网 IP 分配规划 VPC IP 地址空间。
9. 选择 IPv6 VPC CIDR 块。
10. 对于 IPv6 子网 CIDR 块，请为子网选择一个与 VPC CIDR 相同或者更具体的 CIDR。例如，假设 VPC 池 CIDR 为 /50，则可以为子网选择介于 /50 至 /64 之间的网络掩码长度。可能的 IPv6 网络掩码长度介于 /44 和 /64 之间，增量为 /4。
11. 选择创建子网。

使用 AWS CLI 为 VPC 添加子网

使用 create-subnet 命令。

后续步骤

创建子网后，您可以按如下方式对其进行配置：

- 配置路由。然后，您可以创建一个自定义路由表，以及会将流量发送到与 VPC 关联的网关（例如互联网网关）的路由。有关更多信息，请参阅 配置路由表。

- 修改 IP 寻址行为。您可以指定在该子网中启动的实例是将接收公有 IPv4 地址、IPv6 地址或者两者都接收。有关更多信息，请参阅 修改子网的 IP 寻址属性。
- 修改基于资源的名称（RBN）设置。有关更多信息，请参阅 Amazon EC2 实例类型。
- 创建或修改网络 ACL。有关更多信息，请参阅 使用网络访问控制列表控制子网流量。
- 与其他账户共享子网。有关更多信息，请参阅 ???。

将 IPv6 CIDR 块添加到子网或从中删除

您可以向 VPC 中的现有子网关联 IPv6 CIDR 块。此子网当前必须尚未关联任何 IPv6 CIDR 块。

如果不再需要在子网中支持 IPv6，但需要继续使用子网来创建 IPv4 资源并与之通信，则可以移除 IPv6 CIDR 块。

您必须首先将分配给子网中任何实例的任何 IPv6 地址取消分配，然后才能移除 IPv6 CIDR 块。

将 IPv6 CIDR 块添加到子网或从中删除

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Subnets(子网)。
3. 选择您的子网，然后选择 Actions（操作）和 Edit IPv6 CIDRs（编辑 IPv6 CIDR）。
4. 要添加 CIDR，依次选择添加 IPv6 CIDR、VPC CIDR 块，再输入子网 CIDR 块，然后选择与 VPC CIDR 块的网络掩码长度相同或更具体的网络掩码长度。例如，假设 VPC 池 CIDR 为 /50，则可以为子网选择介于 /50 至 /64 之间的网络掩码长度。可能的 IPv6 网络掩码长度介于 /44 和 /64 之间，增量为 /4。
5. 要删除 CIDR，找到 IPv6 CIDR 块并选择删除。
6. 选择保存。

使用 AWS CLI 将 IPv6 CIDR 块关联到子网

使用 associate-subnet-cidr-block 命令。

使用 AWS CLI 取消 IPv6 CIDR 块与子网的关联

使用 disassociate-subnet-cidr-block 命令。

修改子网的 IP 寻址属性

默认情况下，非默认子网的 IPv4 公有寻址属性设置为 `false`，默认子网的此属性设置为 `true`。Amazon EC2 启动实例向导创建的非默认子网属于例外 — 该向导会将此属性设置为 `true`。您可以使用 Amazon VPC 控制台修改此属性。

默认情况下，所有子网的 IPv6 寻址属性都设置为 `false`。您可以使用 Amazon VPC 控制台修改此属性。如果您为子网启用了 IPv6 寻址属性，则在此子网中创建的网络接口会收到此子网范围内的 IPv6 地址。在此子网中启动的实例会在主网络接口上收到一个 IPv6 地址。

您的子网必须具有关联的 IPv6 CIDR 块。

> **Note**
>
> 如果您为子网启用了 IPv6 寻址功能，则只有在您的网络接口或实例是使用 Amazon EC2 API 的 `2016-11-15` 版本或更高版本创建的情况下，您的网络接口或实例才会接收 IPv6 地址。Amazon EC2 控制台使用最新的 API 版本。

修改子网的 IP 寻址行为

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Subnets(子网)。
3. 选择您的子网，然后依次选择 Actions（操作）、Edit subnet settings（编辑子网设置）。
4. 如果选中，则 Enable auto-assign public IPv4 address 复选框会为在所选子网中启动的所有实例请求公有 IPv4 地址。根据需要选中或清除该复选框，然后选择 Save。
5. 如果选中 Enable auto-assign IPv6 address 复选框，则会为在所选子网中创建的所有网络接口请求 IPv6 地址。根据需要选中或清除该复选框，然后选择 Save。

使用 AWS CLI 修改子网属性

使用 modify-subnet-attribute 命令。

子网 CIDR 预留

子网 CIDR 预留是您预留的 IPv4 或 IPv6 地址范围，因此 AWS 不会将这些地址分配到您的网络接口。这使您能够预留用于网络接口的 IPv4 或 IPv6 CIDR 块（也称“前缀”）。

创建子网 CIDR 预留时，您可以指定如何使用预留 IP 地址。以下选项可用：

- 前缀：允许您为单个网络接口分配前缀。有关更多信息，请参阅《Amazon EC2 用户指南》中的为 Amazon EC2 网络接口分配前缀。
- 显式：允许您手动为单个网络接口分配单个 IP 地址。

以下规则适用于子网 CIDR 预留：

- 创建子网 CIDR 预留时，IP 地址范围可以包含正在使用的地址。创建子网预留不会取消分配任何正在使用的 IP 地址。
- 您可以为每个子网预订多个 CIDR 范围。当您在同一 VPC 中预订多个 CIDR 范围时，CIDR 范围不能重叠。
- 如您在子网中为前缀委派预留多个范围，并将前缀委派配置为自动分配时，我们会随机选择要分配到网络接口的 IP 地址。
- 删除子网预留后，AWS 可以将未使用的 IP 地址分配到您的网络接口。删除子网预留不会取消分配任何正在使用的 IP 地址。

有关无类别域间路由（CIDR）表示法的更多信息，请参阅 IP 寻址。

内容

- 通过控制台使用子网 CIDR 预留
- 通过 AWS CLI 使用子网 CIDR 预留

通过控制台使用子网 CIDR 预留

您可以按如下方式创建和管理子网 CIDR 预留。

编辑子网 CIDR 预留

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Subnets（子网）。
3. 选择子网。
4. 选择 CIDR 预留选项卡，以获取有关任何现有子网 CIDR 预留的信息。
5. 要添加或删除子网 CIDR 预留，请选择操作、编辑 CIDR 预留，然后执行以下操作：

- 要添加 IPv4 CIDR 预留，请选择 IPv4、Add IPv4 CIDR reservation（添加 IPv4 CIDR 预留）。选择预留类型，输入 CIDR 范围，然后选择 Add（添加）。
- 要添加 IPv6 CIDR 预留，请选择 IPv6、Add IPv6 CIDR reservation（添加 IPv6 CIDR 预留）。选择预留类型，输入 CIDR 范围，然后选择 Add（添加）。
- 要删除 CIDR 预留，请为子网 CIDR 预留选择删除。

通过 AWS CLI 使用子网 CIDR 预留

您可以使用 AWS CLI 创建和管理子网 CIDR 预留。

任务

- 创建子网 CIDR 预留
- 查看子网 CIDR 预留
- 删除子网 CIDR 预留

创建子网 CIDR 预留

您可以使用 create-subnet-cidr-reservation 创建子网 CIDR 预留。

```
aws ec2 create-subnet-cidr-reservation --subnet-id subnet-03c51e2eEXAMPLE --
reservation-type prefix --cidr 2600:1f13:925:d240:3a1b::/80
```

下面是示例输出。

```
{
    "SubnetCidrReservation": {
        "SubnetCidrReservationId": "scr-044f977c4eEXAMPLE",
        "SubnetId": "subnet-03c51e2ef5EXAMPLE",
        "Cidr": "2600:1f13:925:d240:3a1b::/80",
        "ReservationType": "prefix",
        "OwnerId": "123456789012"
    }
}
```

查看子网 CIDR 预留

您可以使用 get-subnet-cidr-reservations 查看子网 CIDR 预留的详细信息。

```
aws ec2 get-subnet-cidr-reservations --subnet-id subnet-05eef9fb78EXAMPLE
```

删除子网 CIDR 预留

您可以使用 delete-subnet-cidr-reservation 删除子网 CIDR 预留。

```
aws ec2 delete-subnet-cidr-reservation --subnet-cidr-reservation-
id scr-044f977c4eEXAMPLE
```

配置路由表

路由表充当虚拟私有云 (VPC) 的流量控制器。每个路由表都包含一组规则（称为路由），用于确定来自子网或网关的网络流量的指向位置。当您创建 VPC 时，我们也会为 VPC 创建主路由表。您可以为您的 VPC 创建附加路由表，以便对 VPC 的网络路径进行更精细的控制。

您可以使用路由表来指定您的 VPC 可以与哪些网络通信，例如其他 VPC 或本地网络。每个路由指定一个目的地（CIDR 块或前缀列表）和一个目标（例如互联网网关、NAT 网关、VPC 对等连接或 VPN 连接）。流量根据其目的地 IP 地址路由到目标。路由表使您能够创建复杂的网络架构，其中包括公有子网、私有子网、仅 VPN 子网和隔离子网。

内容

- 路由表概念
- 子网路由表
- 网关路由表
- 路由优先级的工作原理
- 示例路由选项
- 为 VPC 创建路由表
- 管理子网路由表
- 替换主路由表
- 使用网关路由表控制进入 VPC 的流量
- 替换或还原本地路由的目标
- 使用 VPC Route Server 在 VPC 中动态路由
- 排查 VPC 中的可达性问题

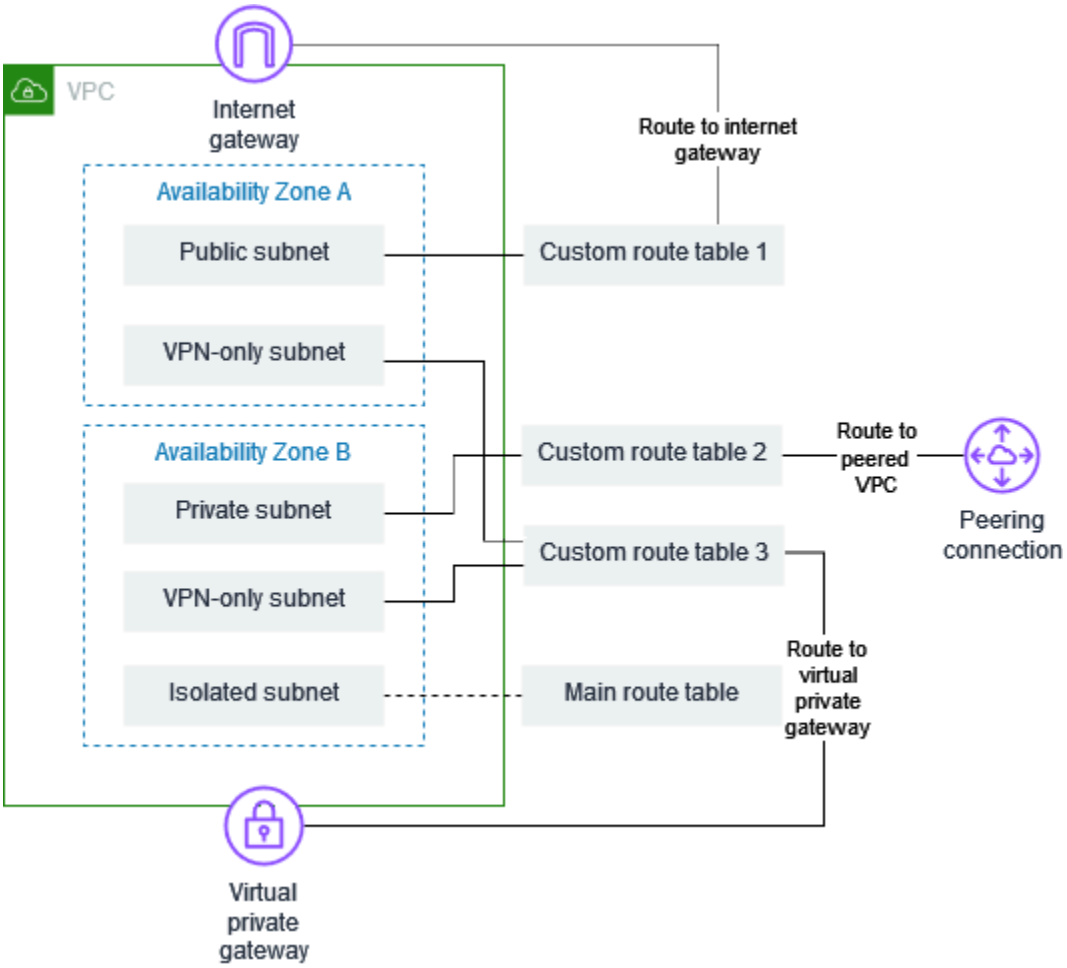
路由表概念

下面是路由表的关键概念：

- 主路由表 — 随 VPC 自动生成的路由表。它控制未与任何其他路由表显式关联的所有子网的路由。
- 自定义路由表 — 您为 VPC 创建的路由表。
- 目的地 — 您希望流量传输到的 IP 地址范围（目的地 CIDR）。例如，具有 CIDR `172.16.0.0/12` 的外部公司网络。
- 目标 — 用于发送目的地流量的网关、网络接口或连接，例如互联网网关。
- 本地路由 — VPC 内通信的默认路由。如果 VPC 同时具有 IPv4 和 IPV6 地址，则存在 IPv4 的本地路由和 IPv6 的本地路由。
- 路由表关联 — 路由表与子网、互联网网关或虚拟私有网关之间的关联。
- 子网路由表 — 与子网关联的路由表。
- 传播–如果您已将虚拟私有网关连接到 VPC 并启用路由传播，我们会自动将 VPN 连接的路由添加到子网路由表。这意味着 VPN 路由无需手动添加或删除。有关更多信息，请参阅《Site-to-Site VPN 用户指南》中的 Site-to-Site VPN 路由选项。
- 网关路由表 — 与互联网网关或虚拟私有网关关联的路由表。
- 边缘关联 — 用于将入站 VPC 流量路由到设备的路由表。您需要将路由表与互联网网关或虚拟私有网关相关联，并将设备的网络接口指定为 VPC 流量的目标。
- 转换网关路由表 – 与转换网关相关联的路由表。有关更多信息，请参阅 Amazon VPC Transit Gateway中的中转网关路由表。
- 本地网关路由表 — 与 Outposts 本地网关相关联的路由表。有关更多信息，请参阅AWS Outposts用户指南中的本地网关。

具有路由表的示例 VPC

下图显示了一个具有五个子网、一个主路由表和三个自定义路由表的 VPC。所有四个路由表都具有本地路由。自定义路由表 1 具有通往互联网网关的路由，并且它与可用区 A 中的公有子网相关联。自定义路由表 2 具有通往对等 VPC 的路由，并且它与可用区 B 中的私有子网相关联。自定义路由表 3 具有通往虚拟私有网关的路由，并且它与两个可用区中的仅 VPN 子网相关联。



子网路由表

VPC 具有隐式路由器，您可以使用路由表来控制网络流量的流向。您的 VPC 中的每个子网必须与一个路由表关联，该路由表控制子网的路由（子网路由表）。您可以将子网与特定路由表显式关联。否则，子网将与主路由表隐式关联。一个子网一次只能与一个路由表关联，但您可以将多个子网与同一子网路由表关联。

内容

- 路线
- 主路由表
- 自定义路由表
- 子网路由表关联

路线

表中的每个路由指定一个目的地和一个目标。例如，要使您的子网能够通过互联网网关访问 Internet，请将以下路由添加到子网路由表中。路由的目的地为 `0.0.0.0/0`，表示所有 IPv4 地址。目标是连接到您的 VPC 的互联网网关。

目标位置	目标
0.0.0.0/0	*igw-id*

IPv4 和 IPv6 的 CIDR 块是分开处理的。例如，目标 CIDR 为 `0.0.0.0/0` 的路由不会自动包括所有 IPv6 地址。您必须为所有 IPv6 地址创建目标 CIDR 为 `::/0` 的路由。

如果您经常在 AWS 资源中引用同一组 CIDR 块，则可以创建客户托管的前缀列表以将它们分组在一起。然后，您可以在路由表条目中将此前缀列表指定为目的地。

每个路由表都包含一个用于在 VPC 内部通信的本地路由。默认情况下，此路由将添加到所有路由表中。如果您的 VPC 有多个 IPv4 CIDR 块，则路由表为每个 IPv4 CIDR 块包含一个本地路由。如果您已将 IPv6 CIDR 块与 VPC 关联，则路由表为 IPv6 CIDR 块包含一个本地路由。您可以根据需要替换或恢复每个本地路由的目标。

规则和注意事项

- 您可以在您的路由表中添加比本地路由更具体的路由。目的地必须匹配 VPC 中子网的整个 IPv4 或 IPv6 CIDR 块。目标必须是 NAT 网关、网络接口或网关负载均衡器端点。
- 如果您的路由表有多个路由，我们使用路由表中与流量匹配的最明确的路由（最长前缀匹配）来判断流量的路由方式。
- 您不能向 IPv4 地址添加与以下范围完全匹配或是其子集的路由：169.254.168.0/22。此范围位于链路本地地址空间内，是专供 AWS 服务使用的保留范围。例如，Amazon EC2 将此范围内的地址用于只能从 EC2 实例访问的服务，例如实例元数据服务（IMDS）和 Amazon DNS 服务器。您可以使用大于但包含 169.254.168.0/22 的 CIDR 块，但是系统不会转发指向 169.254.168.0/22 范围内地址的数据包。
- 您不能向 IPv6 地址添加与以下范围完全匹配或是其子集的路由：fd00:ec2::/32。此范围位于唯一本地地址（ULA）空间内，是专供 AWS 服务使用的保留范围。例如，Amazon EC2 将此范围内的地址用于只能从 EC2 实例访问的服务，例如实例元数据服务（IMDS）和 Amazon DNS 服务器。您可以使用大于但包含 fd00:ec2::/32 的 CIDR 块，但是系统不会转发指向 fd00:ec2::/32 范围内地址的数据包。

- 您可以将中间盒设备添加到 VPC 的路由路径中。有关更多信息，请参阅 the section called “中间盒设备的路由”。

示例

在以下示例中，VPC 具有 IPv4 CIDR 块和 IPv6 CIDR 块。IPv4 和 IPv6 流量是分开处理的，如以下路由表所示。

目标位置	目标
10.0.0.0/16	Local
2001:db8:1234:1a00::/56	Local
172.31.0.0/16	pcx-11223344556677889
0.0.0.0/0	igw-12345678901234567
::/0	eigw-aabbccddee1122334

- Local 路由涵盖了要在 VPC (10.0.0.0/16) 内路由的 IPv4 流量。
- Local 路由涵盖了要在 VPC (2001:db8:1234:1a00::/56) 内路由的 IPv6 流量。
- 172.31.0.0/16 的路由将流量发送到对等连接。
- 所有 IPv4 流量 (0.0.0.0/0) 的路由将流量发送到互联网网关。因此，除了 VPC 内和发送到对等连接的流量外，所有 IPv4 流量都路由到互联网网关。
- 所有 IPv6 流量 (::/0) 的路由将流量发送到仅出口互联网网关。因此，除了 VPC 内和发送到对等连接的流量外，所有 IPv6 流量都路由到仅出口互联网网关。

主路由表

当您创建 VPC 时，它会自动生成主路由表。子网未与显式路由表关联，则预设情况下会使用主路由表。在 Amazon VPC 控制台中的 Route tables（路由表）页面上，通过在 Main（主）列中查找 Yes（是）以查看 VPC 的主路由表。

默认情况下，当您创建非默认 VPC 时，主路由表仅包含本地路由。如果您创建 VPC并选择 NAT 网关，Amazon VPC 会自动将路由添加到网关的主路由表中。

以下规则适用于主路由表：

- 您可以在主路由表中添加、删除和修改路由。
- 您无法删除主路由表。
- 您无法将网关路由表设置为主路由表。
- 您可以通过将自定义路由表与子网关联来替换主路由表。
- 即使某个子网与主路由表已隐式关联，您也可以将它们显式关联。

  在希望更改作为主路由表的表时，您需要执行此操作。当您更改用作主路由表的表时，还会更改其他新子网或所有未与任何其他路由表显式关联的子网的默认设置。有关更多信息，请参阅 替换主路由表。

自定义路由表

默认情况下，每个路由表都包含一个用于在 VPC 内部通信的本地路由。如果您创建 VPC并选择公有子网，Amazon VPC 会创建自定义路由表并添加指向互联网网关的路由。一种保护 VPC 的方法是将主路由表保持原始默认状态。然后，将您创建的各个新子网与您已创建的自定义路由表之一显式关联。这样可以确保您能够明确控制每个子网的流量的路由方式。

您可以在自定义路由表中添加、删除和修改路由。您只能删除没有关联的自定义路由表。

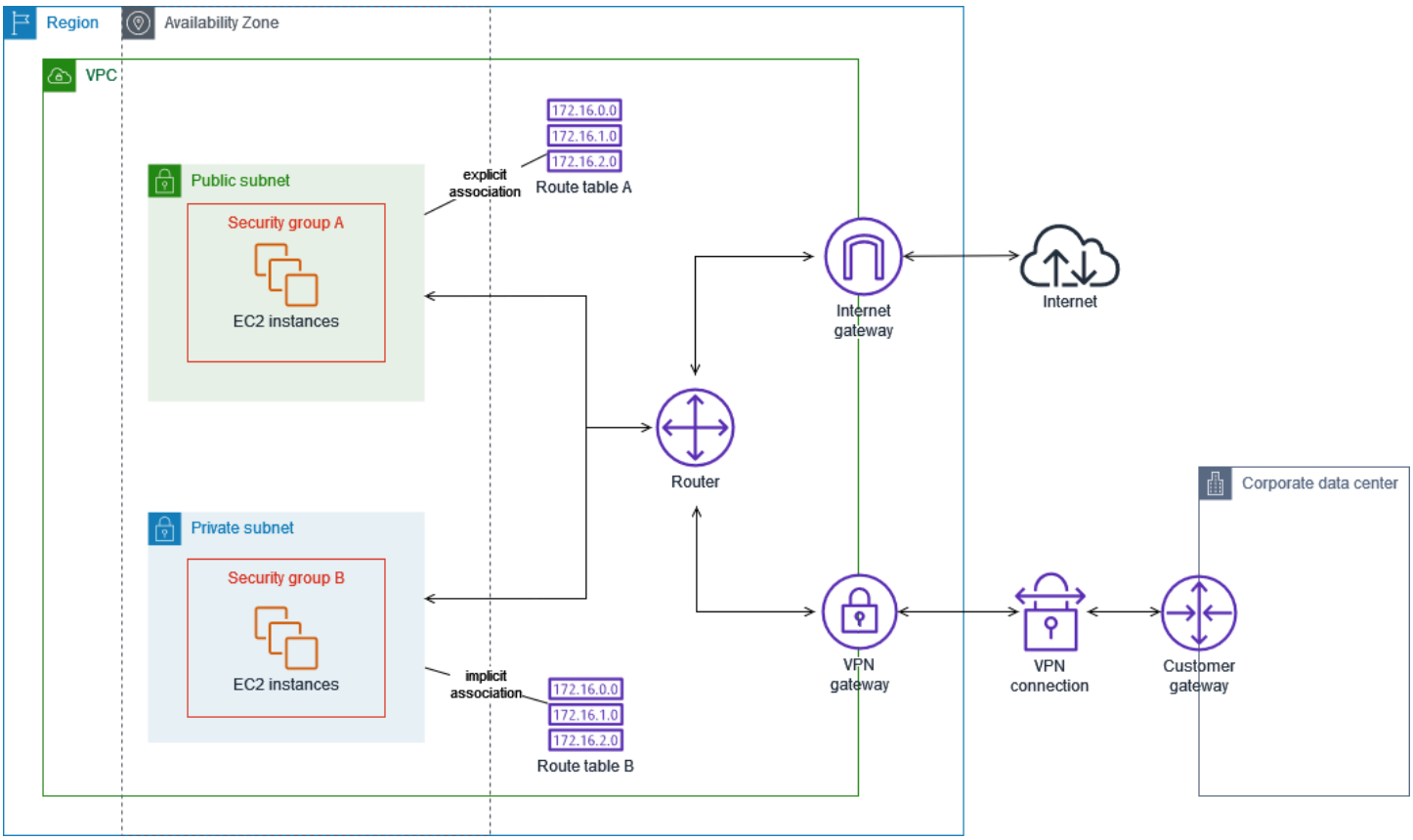
子网路由表关联

您的 VPC 中的每个子网都必须与一个网络 ACL 相关联。子网可以与自定义路由表显式关联，也可以与主路由表隐式或显式关联。有关查看子网和路由表关联的更多信息，请参阅确定显式关联。

与 Outposts 关联的 VPC 中的子网可以有本地网关的额外目标类型。这是与非 Outposts 子网的唯一路由差异。

示例 1：隐式和显式子网关联

下图展示了有互联网网关、虚拟私有网关、以及一个公有子网和仅限 VPN 连接子网的 VPC 的路由。



路由表 A 是与公有子网显式关联的自定义路由表，它有一条将所有流量发送到互联网网关的路由，因此子网成为公有子网。

目标位置	目标
VPC CIDR	本地
0.0.0.0/0	*igw-id*

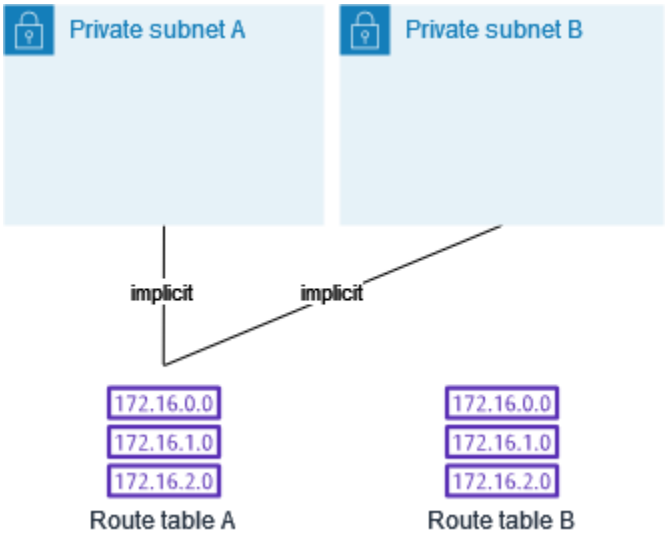
路由表 B 是主路由表，它与私有子网隐式关联。它有一条将所有流量发送到虚拟私有网关的路由，但没有发送到互联网网关的路由，因此子网成为仅限 VPN 的子网。如果您在此 VPC 中创建另一个子网但不关联自定义路由表，由于此路由表是主路由表，该子网也将与其隐式关联。

目标位置	目标
VPC CIDR	本地
0.0.0.0/0	*vgw-id*

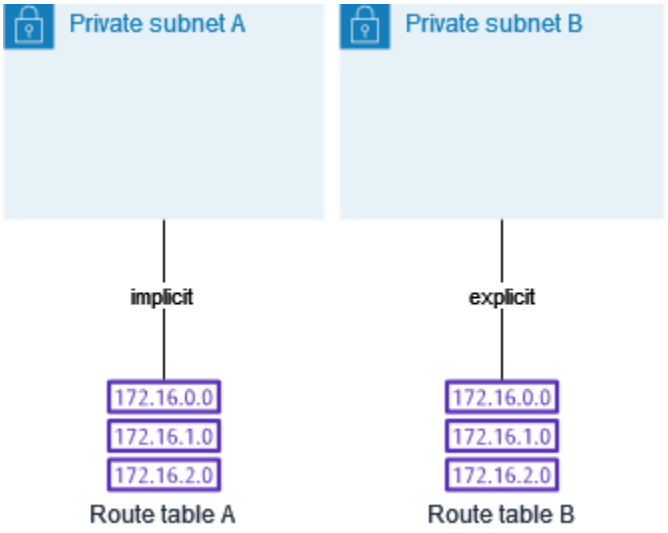
示例 2：替换主路由表

您可能希望更改主路由表。为避免对流量造成任何干扰，我们建议您首先使用自定义路由表测试路由更改。当您满意测试结果之后，可以将主路由表替换为新的自定义路由表。

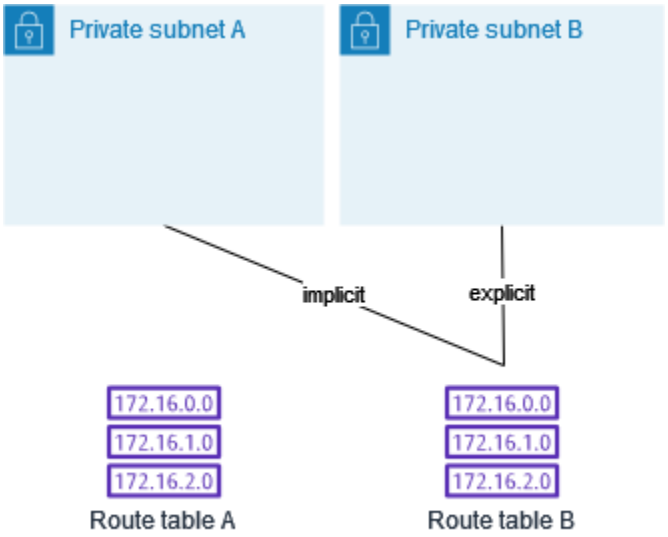
下图显示了两个子网和两个路由表。子网 A 与路由表 A（主路由表）隐式关联。子网 B 与路由表 A 隐式关联。路由表 B（自定义路由表）与这两个子网均未关联。



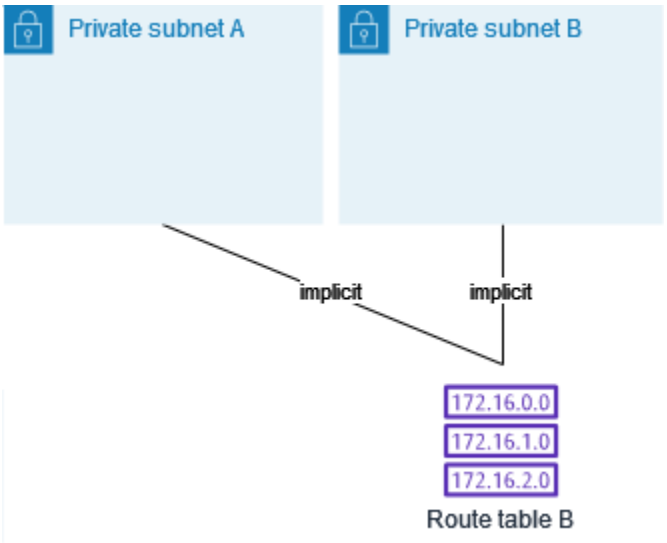
要替换主路由表，请先在子网 B 和路由表 B 之间创建显式关联。测试路由表 B。



在您测试完路由表 B 之后，将其设为主路由表。子网 B 仍与路由表 B 有显式关联。然而子网 A 现在与路由表 B 有隐式关联，因为路由表 B 是新的主路由表。路由表 A 不再与这两个子网关联。



（可选）如果您解除子网 B 与路由表 B 的关联，在子网 B 与路由表 B 之间仍将存在隐式关联。如果您不再需要路由表 A，可以将其删除。



网关路由表

您可以将路由表与互联网网关或虚拟私有网关相关联。当路由表关联到某个网关时，它称为网关路由表。您可以创建网关路由表，以精细控制进入 VPC 的流量的路由路径。例如，对于通过互联网网关进入 VPC 的流量，您可以将流量重定向到 VPC 中的中间盒设备（例如安全设备）来进行拦截。

内容

- 网关路由表路由
- 规则和注意事项

网关路由表路由

与互联网网关关联的网关路由表支持具有以下目标的路由：

- 默认本地路由
- 网关负载均衡器端点
- 中间盒设备的网络接口

与虚拟私有网关关联的网关路由表支持具有以下目标的路由：

- 默认本地路由
- 网关负载均衡器端点
- 中间盒设备的网络接口

当目标是网关负载均衡器端点或网络接口时，允许使用以下目标：

- 您的 VPC 的整个 IPv4 或 IPv6 CIDR 块。在这种情况下，您将替换默认本地路由的目标。
- VPC 中子网的整个 IPv4 或 IPv6 CIDR 块。这是比默认本地路由更明确的路由。

如果您将网关路由表中本地路由的目标更改为 VPC 中的网络接口，则以后可以将其还原为默认 `local` 目标。有关更多信息，请参阅 替换或还原本地路由的目标。

示例

在下面的网关路由表中，流向具有 `172.31.0.0/20` CIDR 块的子网的流量将路由到特定网络接口。流向 VPC 中所有其他子网的流量使用本地路由。

目的地	目标
172.31.0.0/16	本地
172.31.0.0/20	*eni-id*

示例

在以下网关路由表中，本地路由的目标替换为网络接口 ID。流向 VPC 中所有子网的流量将路由到网络接口。

目的地	目标
172.31.0.0/16	*eni-id*

规则和注意事项

如果以下任何情况适用，则无法将路由表与网关相关联：

- 路由表包含的现有路由具有网络接口、网关负载均衡器端点或默认本地路由以外的其他目标。
- 路由表包含的路由指向 VPC 范围之外的 CIDR 块。
- 为路由表启用了路由传播。

此外，还适用以下规则和注意事项：

- 您不能将路由添加到 VPC 范围之外的任何 CIDR 块，包括超出单个 VPC CIDR 块的范围。
- 您只能将 `local`、网关负载均衡器端点或网络接口指定为目标。不能指定任何其他类型的目标，包括单个主机 IP 地址。有关更多信息，请参阅 the section called “示例路由选项”。
- 不能将前缀列表指定为目的地。
- 您不能使用网关路由表来控制或拦截 VPC 外部的流量，例如，流经所连接传输网关的流量。您可以拦截进入您 VPC 的流量，并仅能将其重定向到相同 VPC 中的另一个目标。
- 要确保流量到达您的中间盒设备，必须将目标网络接口连接到正在运行的实例。对于流经互联网网关的流量，目标网络接口还必须具有公有 IP 地址。
- 配置中间盒设备时，请注意设备注意事项。
- 通过中间盒设备路由流量时，来自目标子网的返回流量必须通过同一设备路由。不支持非对称路由。
- 路由表规则适用于离开子网的所有流量。离开子网的流量定义为发往该子网网关路由器 MAC 地址的流量。发往子网中另一个网络接口 MAC 地址的流量使用数据链路（第 2 层）路由而不是网络（第 3 层）路由，因此这些规则不适用于此流量。
- 并非所有 Local Zones 都支持与虚拟私有网关的边缘关联。有关可用区域的更多信息，请参阅《AWS Local Zones 用户指南》中的注意事项。

路由优先级的工作原理

一般来说，我们使用与流量匹配的最明确路由以引导流量。这被称为最长的前缀匹配。如果路由表具有重叠或匹配的路由，则应用其他规则。

以下列表显示了路由优先级摘要，其中包含指向以下部分的链接，包含更多详细信息和示例：

1. 最长前缀（例如，10.10.2.15/32 的优先级高于 10.10.2.0/24）
2. 静态路由（例如 VPC 对等连接和互联网网关连接）
3. 前缀列表路由
4. 传播路由
   a. Direct Connect BGP 路由（动态路由）
   b. VPN 静态路由
   c. VPN BGP 路由（动态路由）（如虚拟专用网关）

最长前缀匹配

到 IPv4 和 IPv6 地址或 CIDR 块的路由彼此独立。我们使用与 IPv4 流量或 IPv6 流量匹配的最明确路由来确定如何路由流量。

下面的子网路由表包含一条指向互联网网关的 IPv4 Internet 流量（`0.0.0.0/0`）路由、一条指向对等连接（`pcx-11223344556677889`）的 `172.31.0.0/16` IPv4 流量路由。来自该子网的目标为 `172.31.0.0/16` IP 地址范围的任意流量均使用对等连接，因为该路由比互联网网关路由更明确。目标设为 VPC (`10.0.0.0/16`) 中的目标的任何流量将被 `local` 路由涵盖，因此将在 VPC 中路由。来自该子网的所有其他流量使用互联网网关。

目的地	目标
10.0.0.0/16	本地
172.31.0.0/16	pcx-11223344556677889
0.0.0.0/0	igw-12345678901234567

静态和动态传播路由的路由优先级

如果您已将一个虚拟私有网关连接到 VPC，并且已启用子网路由表上的路由传播，则代表 Site-to-Site VPN 连接的路由会在您的路由表中自动显示为已传播路由。

如果传播路由的目的地与静态路由的目的地相同，则静态路由优先级更高。以下资源使用静态路由：

- 互联网网关

- NAT 网关
- 网络接口
- 实例 ID
- 网关 VPC 端点
- Transit Gateway
- VPC 对等连接
- 网关负载均衡器端点

有关更多信息，请参阅《AWS Site-to-Site VPN 用户指南》中的路由表和 VPN 路由优先级。

以下示例路由表具有指向互联网网关的静态路由和指向虚拟私有网关的传播路由。这两条路由的目的地均为 `172.31.0.0/24`。由于通往互联网网关的静态路由优先级更高，因此所有到 `172.31.0.0/24` 的流量被路由到互联网网关。

目标位置	目标	传播
10.0.0.0/16	本地	否
172.31.0.0/24	vgw-11223344556677889	是
172.31.0.0/24	igw-12345678901234567	否

前缀列表的路由优先级

如果路由表引用前缀列表，则以下规则适用：

- 如果路由表中包含的传播路由与引用前缀列表的路由匹配，则引用前缀列表的路由优先。请注意，如果路由出现重叠，无论它们是传播路由、静态路由还是引用前缀列表的路由，更具体的路由始终优先。
- 如果路由表引用多个前缀列表，而这些前缀列表具有到不同目标的重叠的 CIDR 块，则我们会随机选择哪条路由优先。此后，同一路由将始终优先。

示例路由选项

以下主题介绍了您的 VPC 中的特定网关或连接的路由。

内容

- 路由到互联网网关
- 路由到 NAT 设备
- 路由到虚拟私有网关
- 路由到 AWS Outposts 本地网关
- 路由到 VPC 对等连接
- 路由到网关 VPC 端点
- 路由到仅出口互联网网关
- 中转网关的路由
- 中间盒设备的路由
- 使用前缀列表进行路由
- 路由到网关负载均衡器端点

路由到互联网网关

您可以通过向互联网网关添加子网路由表中的路由，使子网成为公有子网。为此，请创建一个互联网网关并将其附加到您的 VPC，然后添加一个目的地为 `0.0.0.0/0` (对于 IPv4 流量) 或 `::/0` (对于 IPv6 流量) 且目标为互联网网关 ID (`igw-xxxxxxxxxxxxxxxxx`) 的路由。

目的地	目标
0.0.0.0/0	*igw-id*
::/0	*igw-id*

有关更多信息，请参阅 使用互联网网关为 VPC 启用互联网访问。

路由到 NAT 设备

要使私有子网中的实例能够连接到 Internet，您可以创建 NAT 网关或在公有子网中启动 NAT 实例。然后为私有子网的路由表添加路由，将 IPv4 Internet 流量 (`0.0.0.0/0`) 路由到 NAT 设备。

目的地	目标
0.0.0.0/0	*nat-gateway-id*

您还可以创建到其他目标的更明确的路由，以避免因使用 NAT 网关而产生的不必要的数据处理费用，或者需要专门路由特定流量。在以下示例中，Amazon S3 流量（pl-xxxxxxxx，前缀列表,包含特定区域中 Amazon S3 的 IP 地址范围）路由到网关 VPC 端点，并且 10.25.0.0/16 流量路由到 VPC 对等连接。这些 IP 地址范围比 0.0.0.0/0 更具体。当实例向 Amazon S3 或对等 VPC 发送流量时，流量将发送到网关 VPC 端点或 VPC 对等连接。所有其他流量发送到 NAT 网关。

目的地	目标
0.0.0.0/0	*nat-gateway-id*
pl-*xxxxxxxx*	*vpce-id*
10.25.0.0/16	*pcx-id*

有关更多信息，请参阅 NAT 设备。

路由到虚拟私有网关

您可以使用AWS Site-to-Site VPN 连接来支持 VPC 中的实例与您自己的网络进行通信。为此，请创建虚拟私有网关并附加到您的 VPC。然后在子网路由表中添加路由，其目的地为您的网络，目标为虚拟私有网关 (vgw-xxxxxxxxxxxxxxxxx)。

目的地	目标
10.0.0.0/16	*vgw-id*

然后，您可以创建和配置 Site-to-Site VPN 连接。有关更多信息，请参阅《AWS Site-to-Site VPN 用户指南》中的什么是 AWS Site-to-Site VPN？和路由表和 VPN 路由优先级。

虚拟私有网关上的 Site-to-Site VPN 连接不支持 IPv6 流量。但是，我们支持通过虚拟私有网关路由到 AWS Direct Connect 连接的 IPv6 流量。有关更多信息，请参阅 AWS Direct Connect 用户指南。

路由到 AWS Outposts 本地网关

本节介绍用于路由到 AWS Outposts 本地网关的路由表配置。

内容

- 启用 Outpost 子网与本地网络之间的流量
- 在不同 Outposts 的同一 VPC 中的子网之间启用流量

启用 Outpost 子网与本地网络之间的流量

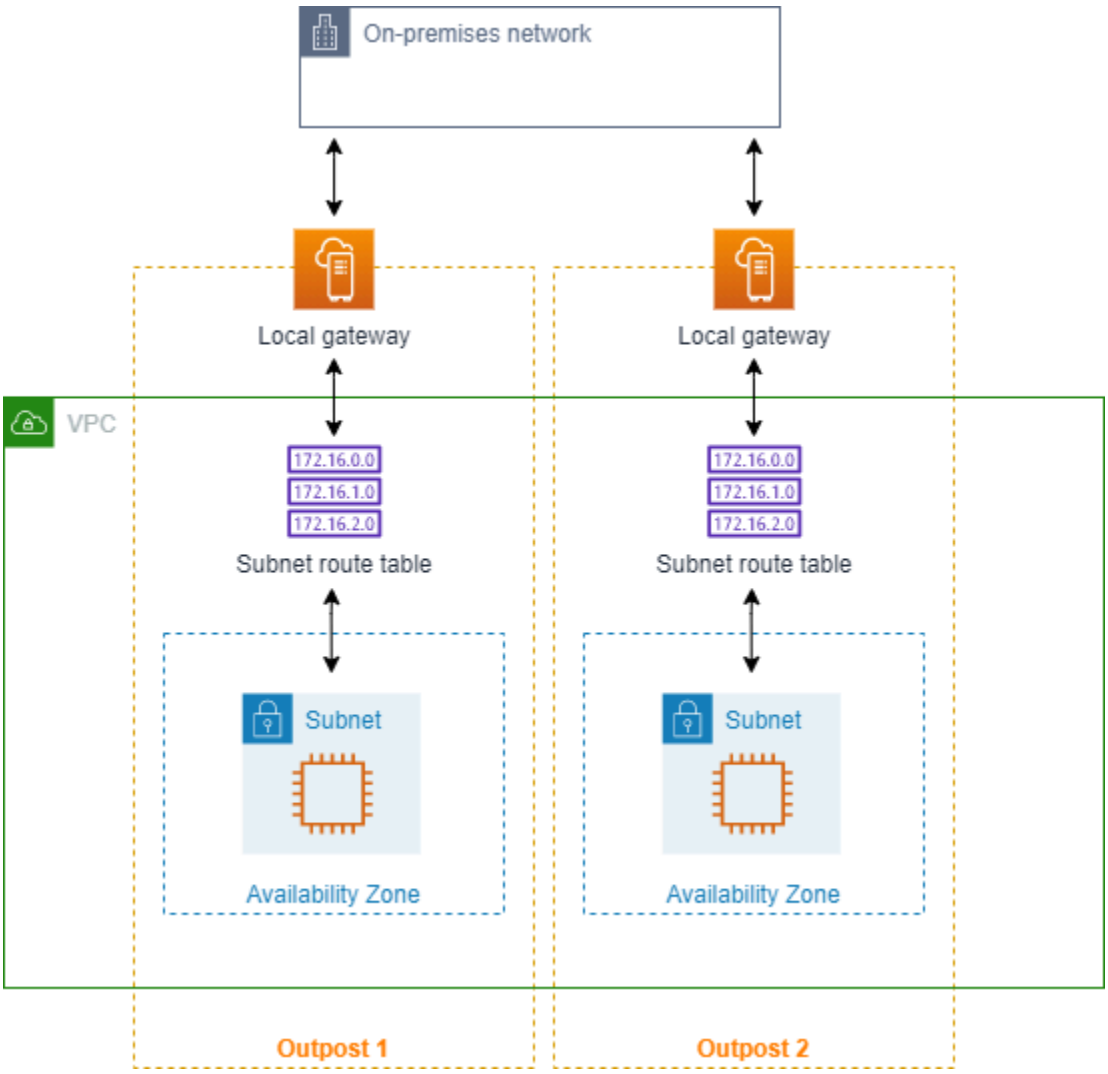
与 AWS Outposts 关联的 VPC 中的子网可以有本地网关的额外目标类型。考虑您希望将目的地地址为 192.168.10.0/24 的本地网关路由流量到客户网络的情况。要执行此操作，请添加具有目的地网络和本地网关目标 (`lgw-xxxx`) 的以下路由。

目的地	目标
192.168.10.0/24	*lgw-id*

在不同 Outposts 的同一 VPC 中的子网之间启用流量

您可以使用 Outpost 本地网关和本地网络，在不同 Outpost 的同一 VPC 中的子网之间建立通信。

借助此功能，您可以通过在锚定到不同可用区的 Outposts 机架之间建立连接，为在 Outposts 机架上运行的本地应用程序构建类似于多可用区（AZ）架构的架构。



要启用此功能，请向 Outpost 机架子网路由表添加一个路由，该路由应比路由表中的本地路由更为具体，且目标类型应为本地网关。路由目标必须匹配另一个 Outpost 中 VPC 中子网的整个 IPv4 块。对所有需要进行通信的 Outpost 子网重复此配置。

Important

- 要使用此功能，必须使用直接 VPC 路由。不得使用客户自有 IP 地址。
- Outposts 本地网关所连接到的本地网络必须具有所需路由，以便子网能够相互访问。
- 如果要对子网中的资源使用安全组，则必须使用包含 IP 地址范围作为 Outpost 子网中的源或目标的规则。不得使用安全组 ID。
- 现有 Outposts 机架可能需要更新，才能支持在多个 Outposts 之间进行 VPC 内部通信。如果此功能并不适用，请联系 AWS Support。

Example 示例

对于 CIDR 为 10.0.0.0/16 的 VPC、CIDR 为 10.0.1.0/24 的 Outpost 1 子网，以及 CIDR 为 10.0.2.0/24 的 Outpost 2 子网，Outpost 1 子网的路由表条目将如下所示：

目标位置	目标
10.0.0.0/16	本地
10.0.2.0/24	*lgw-1-id*

Outpost 2 子网的路由表条目将如下所示：

目标位置	目标
10.0.0.0/16	本地
10.0.1.0/24	*lgw-2-id*

路由到 VPC 对等连接

VPC 对等连接是两个 VPC 之间的网络连接，通过此连接，您可以使用私有 IPv4 地址在这两个 VPC 之间路由流量。任何一个 VPC 中的实例都可以彼此通信，就像它们属于同一网络中一样。

要在 VPC 对等连接中的 VPC 之间实现流量路由，您必须将一个路由添加到指向 VPC 对等连接的一个或多个子网路由表。这允许您访问对等连接中其他 VPC 的全部或部分 CIDR 块。同样，另一个 VPC 的拥有者必须将一个路由添加到其子网路由表，以将流量路由回您的 VPC。

例如，您在具有以下信息的两个 VPC 之间具有 VPC 对等连接 (pcx-11223344556677889)：

- VPC A：CIDR 块为 10.0.0.0/16
- VPC B：CIDR 块为 172.31.0.0/16

要启用 VPC 之间的流量并允许访问任一 VPC 的整个 IPv4 CIDR 块，VPC A 的路由表的配置如下所示。

目的地	目标
10.0.0.0/16	本地
172.31.0.0/16	pcx-11223344556677889

VPC B 的路由表的配置如下所示。

目的地	目标
172.31.0.0/16	本地
10.0.0.0/16	pcx-11223344556677889

您的 VPC 对等连接也可以支持 VPC 中实例之间的 IPv6 通信，前提是已启用 VPC 和实例进行 IPv6 通信。要在 VPC 之间启用 IPv6 流量路由，您必须向路由表中添加一条指向 VPC 对等连接的路由，以访问对等 VPC 的全部或部分 IPv6 CIDR 块。

例如，仍使用上面的 VPC 对等连接 (pcx-11223344556677889)，假设 VPC 具有以下信息：

- VPC A：IPv6 CIDR 块为 2001:db8:1234:1a00::/56
- VPC B：IPv6 CIDR 块为 2001:db8:5678:2b00::/56

要通过 VPC 对等连接启用 IPv6 通信，请将以下路由添加到 VPC A 的子网路由表中。

目的地	目标
10.0.0.0/16	本地
172.31.0.0/16	pcx-11223344556677889
2001:db8:5678:2b00::/56	pcx-11223344556677889

将以下路由添加到 VPC B 的路由表中：

目的地	目标
172.31.0.0/16	本地
10.0.0.0/16	pcx-11223344556677889
2001:db8:1234:1a00::/56	pcx-11223344556677889

有关 VPC 对等连接的更多信息，请参阅 Amazon VPC 对等连接指南。

路由到网关 VPC 端点

使用网关 VPC 端点可以在您的 VPC 和其他 AWS 服务之间创建私有连接。创建网关端点时，您指定 VPC 中由该网关端点使用的子网路由表。路由会自动添加到每个路由表中，这些路由表的目的地指定服务的前缀列表 ID (`pl-xxxxxxxx`)，目标具有相应端点 ID (`vpce-xxxxxxxxxxxxxxxxx`)。您无法显式删除或修改端点路由，但可更改端点所使用的路由表。

有关端点路由的更多信息以及对到 AWS 服务的路由的影响，请参阅网关端点路由。

路由到仅出口互联网网关

您可以为 VPC 创建仅出口互联网网关，以允许私有子网中的实例发起到 Internet 的出站通信，但阻止 Internet 发起与这些实例的连接。仅出口互联网网关只适用于 IPv6 流量。要为仅出口互联网网关配置路由，请为将 IPv6 Internet 流量 (`::/0`) 路由到仅出口互联网网关的私有子网路由表添加路由。

目的地	目标
::/0	*eigw-id*

有关更多信息，请参阅 使用仅出口互联网网关允许出站 IPv6 流量。

中转网关的路由

将 VPC 附加到中转网关时，您需要向子网路由表添加路由，以使流量通过中转网关进行路由。

考虑以下场景：您有三个 VPC 附加到中转网关。在该方案中，所有挂载与中转网关路由表相关联，并传播到中转网关路由表。因此，所有挂载都可以将数据包路由到彼此，而将中转网关用作简单第 3 层 IP 集线器。

例如，您有两个 VPC，其中包含以下信息：

- VPC A：10.1.0.0/16，附加 ID tgw-attach-11111111111111111
- VPC B：10.2.0.0/16，连接 ID tgw-attach-22222222222222222

要启用 VPC 之间的流量并允许访问中转网关，VPC A 路由表的配置如下所示。

目的地	目标
10.1.0.0/16	本地
10.0.0.0/8	*tgw-id*

以下是 VPC 挂载的中转网关路由表条目的示例。

目的地	目标
10.1.0.0/16	tgw-attach-11111111111111111
10.2.0.0/16	tgw-attach-22222222222222222

有关中转网关路由表的更多信息，请参阅 Amazon VPC Transit Gateway 中的路由。

中间盒设备的路由

您可以将中间盒设备添加到 VPC 的路由路径中。以下是可能使用案例：

- 拦截通过互联网网关或虚拟私有网关进入 VPC 的流量，方法是将其引导到 VPC 中的中间盒设备。您可以使用中间盒路由向导，让AWS自动为网关、中间盒设备和目标子网配置相应的路由表。有关更多信息，请参阅 the section called “中间盒路由向导”。
- 将两个子网之间的流量定向到中间盒设备。您可以通过为与另一个子网的子网 CIDR 匹配的子网路由表创建路由，并将网关负载均衡器端点、NAT 网关、Network Firewall 端点或设备的网络接口指定为目标来完成此操作。或者，要将所有流量从子网重新导向到任何其他子网，请将本地路由的目标替换为网关负载均衡器端点、NAT 网关或网络接口。

您可以配置设备来满足要求。例如，您可以配置筛选所有流量的安全设备或 WAN 加速设备。设备作为 Amazon EC2 实例部署在 VPC 的子网中，并由子网中的弹性网络接口（网络接口）呈现。

如果您为目标子网路由表启用路由传播，请注意路由优先级。我们将确定最具体路由的优先级，如果路由匹配，静态路由的优先级将高于传播路由。检查您的路由，确保正确路由了流量，并且在启用或禁用路由传播时不会产生意外后果（例如，支持巨帧的 AWS Direct Connect 连接需要路由传播）。

要将入站 VPC 流量路由到设备，您需要将路由表与互联网网关或虚拟私有网关相关联，并将设备的网络接口指定为 VPC 流量的目标。有关更多信息，请参阅 网关路由表。您还可以将出站流量从您的子网路由到另一个子网中的中间设备。

有关中间盒路由示例，请参阅中间盒场景。

内容

- 设备注意事项
- 在网关和设备之间路由流量
- 将子网间流量路由到设备

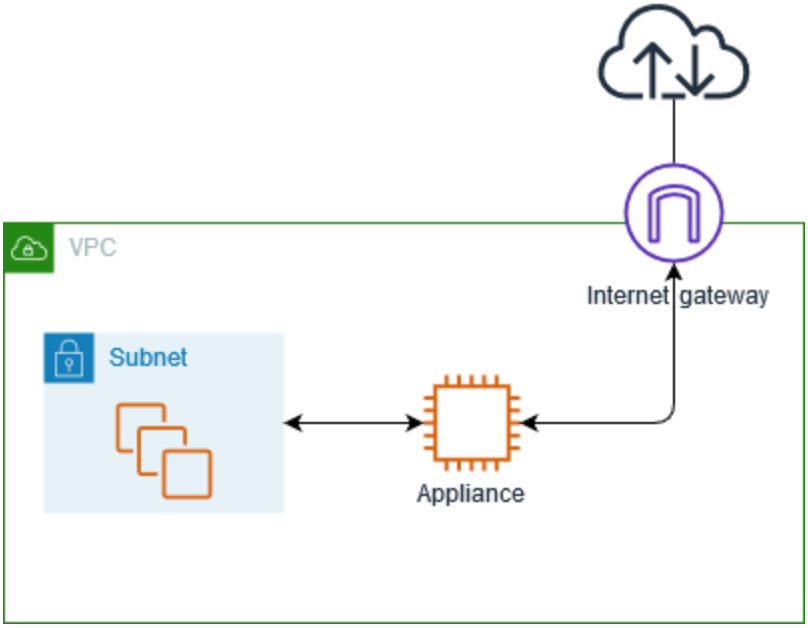
设备注意事项

您可以从 AWS Marketplace 选择第三方设备，也可以配置自己的设备。创建或配置设备时，请注意以下事项：

- 设备必须在指向源流量或目标流量的单独子网中配置。
- 您必须禁用设备上的源/目标检查。有关更多信息，请参阅《Amazon EC2 用户指南》中的更改源或目标检查。
- 您不能通过设备在同一子网中的主机之间路由流量。
- 设备不必执行网络地址转换 (NATI)。
- 您可以在您的路由表中添加比本地路由更具体的路由。您可以使用更具体的路由将 VPC（东-西流量）内子网之间的流量重新导向到中间盒设备。路由目的地必须匹配 VPC 中子网的整个 IPv4 或 IPv6 CIDR 块。
- 要拦截 IPv6 流量，请确保您的 VPC、子网和设备支持 IPv6。

在网关和设备之间路由流量

要将入站 VPC 流量路由到设备，您需要将路由表与互联网网关或虚拟私有网关相关联，并将设备的网络接口指定为 VPC 流量的目标。在以下示例中，VPC 具有互联网网关、设备和子网，其中包含实例。来自互联网的流量通过设备进行路由。



将此路由表与互联网网关或虚拟私有网关相关联。第一个条目是本地路由。第二个条目将流向子网的 IPv4 流量发送到设备的网络接口。这是比默认本地路由更明确的路由。

目标位置	目标
VPC CIDR	本地
## CIDR	*###### ID*

或者，您可以将本地路由的目标替换为设备的网络接口。您可以执行此操作以确保所有流量自动路由到设备，包括流向您以后添加到 VPC 的子网的流量。

目标位置	目标
VPC CIDR	*###### ID*

要将流量从您的子网路由到另一个子网中的设备，请向您的子网路由表添加将流量路由到设备网络接口的路由。目的地的具体程度必须低于本地路由的目的地。例如，对于流向 Internet 的流量，请为目的地指定 `0.0.0.0/0`（所有 IPv4 地址）。

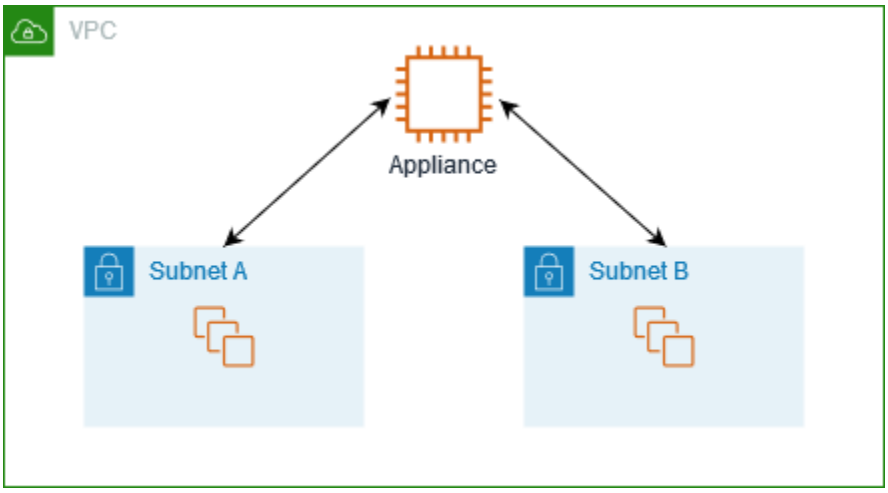
目的地	目标
VPC CIDR	本地
0.0.0.0/0	*###### ID*

然后，在与设备子网关联的路由表中，添加将流量发送回互联网网关或虚拟私有网关的路由。

目标位置	目标
VPC CIDR	本地
0.0.0.0/0	*igw-id*

将子网间流量路由到设备

您可以将流向特定子网的流量路由到设备的网络接口。在以下示例中，VPC 包含两个子网和一个设备。通过设备在子网之间路由的流量。



安全组

当您通过中间盒设备在不同子网中的实例之间路由流量时，这两个实例的安全组必须允许流量在实例之间流动。每个实例的安全组必须引用另一个实例的私有 IP 地址或包含另一个实例的子网的 CIDR 范围作为源。如果您引用另一个实例的安全组作为源，则安全组不允许流量在实例之间流动。

路由

以下是子网 A 的示例路由表。第一个条目允许 VPC 中的实例在彼此之间进行通信。第二个条目将从子网 A 到子网 B 的所有流量路由到设备的网络接口。

目标位置	目标
VPC CIDR	本地
## B CIDR	*###### ID*

以下是子网 B 的示例路由表。第一个条目允许 VPC 中的实例在彼此之间进行通信。第二个条目将从子网 B 到子网 A 的所有流量路由到设备的网络接口。

目标位置	目标
VPC CIDR	本地
## A CIDR	*###### ID*

或者，您可以将本地路由的目标替换为设备的网络接口。您可以执行此操作以确保所有流量自动路由到设备，包括流向您以后添加到 VPC 的子网的流量。

目标位置	目标
VPC CIDR	*###### ID*

使用前缀列表进行路由

如果您经常在 AWS 资源中引用同一组 CIDR 块，则可以创建客户托管的前缀列表以将它们分组在一起。然后，您可以在路由表条目中将此前缀列表指定为目的地。您可以稍后添加或删除前缀列表的条目，而无需更新路由表。

例如，您具有一个包含多个 VPC 连接的中转网关。VPC 必须能够与具有以下 CIDR 块的两个特定 VPC 连接进行通信：

- 10.0.0.0/16
- 10.2.0.0/16

创建包含这两个条目的前缀列表。在子网路由表中，创建一条路由，并将前缀列表指定为目的地，将中转网关指定为目标。

目的地	目标
172.31.0.0/16	本地
pl-123abc123abc123ab	*tgw-id*

前缀列表的最大条目数与路由表中的条目数相等。

路由到网关负载均衡器端点

网关负载均衡器使您能够将流量分配到虚拟设备队列，例如防火墙。您可以创建网关负载均衡器，配置网关负载均衡器端点服务，然后在您的 VPC 中创建网关负载均衡器端点以将其连接到该服务。

要将流量路由到网关负载均衡器（例如，用于安全检查），请在路由表中将网关负载均衡器端点指定为目标。

有关网关负载均衡器后面的安全设备的示例，请参阅the section called “使用安全设备检查流量”。

要在路由表中指定网关负载均衡器端点，请使用 VPC 端点的 ID。例如，要将 10.0.1.0/24 的流量路由到网关负载均衡器端点，请添加以下路由。

目标位置	目标
10.0.1.0/24	*vpc-endpoint-id*

使用 Gateway Load Balancer 端点作为目标时，不能将前缀列表指定为目的地。如果您尝试创建或替换以 VPC 端点为目标的前缀列表路由，则会收到错误消息：“无法创建或替换以 VPC 端点为目标的前缀列表路由。”

有关更多信息，请参阅网关负载均衡器。

为 VPC 创建路由表

完成以下任务来为您的 VPC 创建和配置自定义路由表。默认情况下，您的新路由表包含允许在 VPC 内通信的本地路由。您可以添加路由，根据目的地 IP 地址范围将网络流量定向到特定目标。

若要对特定子网应用路由表路由，您必须将路由表与子网关联。一个路由表可以与多个子网关联。但是，子网一次只能与一个路由表关联。任何未与路由表显式关联的子网都默认与主路由表隐式关联。

您可以解除子网与路由表的关联。在将子网与其他路由表关联前，它与主路由表是隐式关联的。

> **Note**
>
> 您可以为每个 VPC 创建的路由表数量存在配额。您可以为每个路由表添加的路由数量也有配额。有关更多信息，请参阅 Amazon VPC 配额。

任务

- 创建路由表
- 向路由表添加路由
- 将子网与路由表关联

创建路由表

使用控制台创建路由表

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Route tables（路由表）。
3. 选择创建路由表。
4. （可选）对于 Name（名称），为您的路由表输入名称。
5. 对于 VPC，选择您的 VPC。
6. （可选）若要添加标签，请选择 Add new tag（添加新标签），然后输入标签键和标签值。
7. 选择创建路由表。

使用 AWS CLI 创建路由表

使用 create-route-table 命令。

向路由表添加路由

使用控制台向路由表添加路由

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Route tables（路由表），然后选择路由表。
3. 依次选择 Actions（操作）、Edit routes（编辑路由）。
4. 选择 Add route（添加路由）。
5. 对于目的地，请输入以下内容之一：
   - IP 地址范围 - 例如 192.168.0.0/16
   - 单个 IP 地址 - 例如 192.168.10.1/32
   - 前缀列表的 ID - 例如，pl-0abcdef1234567890
6. 对于目标，选择一种资源类型（例如，网络接口），然后输入该资源的 ID（例如，eni-11223344556677889）。
7. 选择保存更改。

使用 AWS CLI 向路由表添加路由

使用 create-route 命令。

将子网与路由表关联

使用控制台将路由表与子网关联

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Route tables（路由表），然后选择路由表。
3. 在 Subnet associations (子网关联) 选项卡上，选择 Edit subnet associations (编辑子网关联)。
4. 选中要与路由表关联的子网的复选框。
5. 选择 Save associations (保存关联)。

使用 AWS CLI 将子网与路由表关联或取消关联

- associate-route-table

- disassociate-route-table

管理子网路由表

使用以下过程通过路由表管理 VPC 路由。

任务

- 确定子网的路由表
- 确定显式关联
- 添加、修改和移除路由
- 启用或禁用路由传播
- 更改子网的路由表

确定子网的路由表

可通过在 Amazon VPC 控制台中查看子网的详细信息，判断该子网与哪个路由表关联。

使用控制台确定子网的路由表

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Subnets（子网）。
3. 选择子网。
4. 选择 Route table（路由表）选项卡可查看路由表 ID 及其路由的相关信息。如需判断是否与主路由表存在关联以及是否为显式关联，请参见 确定显式关联。

确定显式关联

您可以确定哪些子网或网关与路由表显式关联以及存在关联的数量。

主路由表可以有显式和隐式子网关联。自定义路由表只有显式关联。

未与任何路由表建立显式关联的子网都与主路由表有隐式关联。您可以在子网与主路由表间建立显式关联。有关您可能会这样做的原因的示例，请参阅替换主路由表。

使用控制台确定显式关联的子网

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。

2. 在导航窗格中，选择 Route tables（路由表）。
3. 检查 Explicit subnet association（显式子网关联）列以确定显式关联的子网，检查 Main（主）列以确定是否为主路由表。
4. 选择路由表并选择 Subnet associations（子网关联）选项卡。
5. Explicit subnet associations（显式子网关联）下的子网与路由表显式关联。Subnets without explicit associations（没有显式关联的子网）下的子网与路由表属于同一 VPC，但未与任何路由表关联，因此它们与该 VPC 的主路由表隐式关联。

使用控制台确定显式关联的网关

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Route tables（路由表）。
3. 选择路由表并选择 Edge associations（边缘关联）选项卡。

使用 AWS CLI 描述一个或多个路由表并查看其关联

使用 describe-route-tables 命令。

添加、修改和移除路由

您可以添加、修改和删除路由表的路由。

有关使用 Site-to-Site VPN 连接的静态路由的更多信息，请参阅《AWS Site-to-Site VPN 用户指南》中的编辑 Site-to-Site VPN 连接的静态路由。

注意事项

- 您只能修改已添加的路由。
- 如果修改或删除路由，使用这些路由的现有连接会受到影响，而使用其他路由的连接将不受影响。
- 您可以为每个路由表添加的路由数量有配额。有关更多信息，请参阅 Amazon VPC 配额。

使用控制台更新路由表的路由

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Route tables（路由表），然后选择路由表。
3. 依次选择 Actions (操作)、Edit routes (编辑路由)。

4. 要添加路由，请选择添加路由。对于目的地，请输入 IP 地址范围、单个 IP 地址或前缀列表的 ID。对于目标，请选择资源类型，然后输入资源的 ID。
5. 要修改路由，请输入新的目的地 CIDR 块或前缀列表 ID，然后选择目标。
6. 要删除路由，请选择 Remove（删除）。
7. 选择保存更改。

使用 AWS CLI 更新路由表的路由

如果您使用命令行工具或 API 添加路由，则目的地 CIDR 块将自动修改为其规范形式。例如，如果您为 CIDR 块指定 `100.68.0.18/18`，我们将创建一个路径，其目标 CIDR 块为 `100.68.0.0/18`。

- create-route
- replace-route
- delete-route

启用或禁用路由传播

路由传播允许虚拟私有网关自动将路由传播到路由表。这意味着 VPN 路由无需手动添加或删除。

要完成此过程，您必须具有虚拟私有网关。

有关更多信息，请参阅《Site-to-Site VPN 用户指南》中的 Site-to-Site VPN 路由选项。

使用控制台启用或禁用路由传播

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Route tables（路由表），然后选择路由表。
3. 依次选择 Actions (操作) 和 Edit route propagation (编辑路由传播)。
4. 选择或清除虚拟专用网关旁边的启用复选框。
5. 选择保存。

使用 AWS CLI 启用或禁用路由传播

- enable-vgw-route-propagation
- disable-vgw-route-propagation

更改子网的路由表

更改子网的路由表关联

当您更改路由表时，将删除子网中的现有连接，除非新路由表中包含相同流量到同一目标的路由。

使用控制台更改子网路由表关联

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Subnets（子网），然后选择子网。
3. 从 Route table（路由表）选项卡，选择 Edit route table association（编辑路由表关联）。
4. 对于 Route table ID（路由表 ID），选择新的路由表。
5. 选择保存。

使用 AWS CLI 更改与子网关联的路由表

使用 replace-route-table-association 命令。

替换主路由表

本节旨在介绍如何更改您 VPC 中作为主路由表的路由表。

使用控制台替换主路由表

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Route tables（路由表），然后选择新的主路由表。
3. 选择 Actions（操作）、Set main route table（设置主路由表）。
4. 提示进行确认时，输入 **`set`**，然后选择 OK（确认）。

使用 AWS CLI 替换主路由表

- 使用 replace-route-table-association 命令。

以下步骤描述如何删除子网与主路由表之间的显式关联。结果是在子网和主路由之间生成隐式关联。这个步骤与解除任何子网与任何路由表的步骤相同。

删除与主路由表的显式关联

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Route tables（路由表），然后选择路由表。
3. 从 Subnet associations（子网关联）选项卡，选择 Edit subnet associations（编辑子网关联）。
4. 清除子网的复选框。
5. 选择 Save associations (保存关联)。

使用网关路由表控制进入 VPC 的流量

要使用网关路由表控制进入 VPC 的流量，您可以将互联网网关或虚拟私有网关与路由表关联或取消关联。有关更多信息，请参阅 网关路由表。

使用控制台将网关与路由表关联或取消关联

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Route tables（路由表），然后选择路由表。
3. 从 Edge associations（边缘关联）选项卡，选择 Edit edge associations（编辑边缘关联）。
4. 选中或取消选中网关的复选框。
5. 选择保存更改。

使用 AWS CLI 将网关与路由表关联

使用 associate-route-table 命令。以下示例将指定的路由表与指定的互联网网关关联。

```
aws ec2 associate-route-table
    --route-table-id rtb-01234567890123456 \
    --gateway-id igw-11aa22bb33cc44dd1
```

使用 AWS CLI 取消网关与路由表的关联

使用 disassociate-route-table 命令。指定路由表与网关的关联的 ID。

```
aws ec2 disassociate-route-table \
    --association-id rtbassoc-0abcdef1234567890
```

替换或还原本地路由的目标

您可以更改默认本地路由的目标。如果您替换本地路由的目标，以后可以将其恢复为默认 `local` 目标。如果您的 VPC 有多个 CIDR 块，则路由表会有多个本地路由，每个 CIDR 块一个。您可以根据需要替换或恢复各个本地路由的目标。

使用控制台替换本地路由

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Route tables（路由表），然后选择路由表。
3. 从 Routes（路由）选项卡，选择 Edit routes（编辑路由）。
4. 对于本地路由，清除 Target（目标），然后选择一个新目标。
5. 选择保存更改。

使用控制台还原本地路由的目标

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Route tables（路由表），然后选择路由表。
3. 依次选择 Actions (操作)、Edit routes (编辑路由)。
4. 对于路由，清除 Target（目标），然后选择 local（本地）。
5. 选择保存更改。

使用 AWS CLI 替换本地路由的目标

使用 replace-route 命令。以下示例将本地路由的目标替换为指定的网络接口。

```
aws ec2 replace-route \
    --route-table-id rtb-01234567890123456 \
    --destination-cidr-block 10.0.0.0/16 \
    --network-interface-id eni-11223344556677889
```

使用 AWS CLI 还原本地路由的目标

以下示例还原指定路由表的本地目标。

```
aws ec2 replace-route \
    --route-table-id rtb-01234567890123456 \
```

```
    --destination-cidr-block 10.0.0.0/16 \
    --local-target
```

使用 VPC Route Server 在 VPC 中动态路由

Amazon VPC Route Server 简化了部署在 VPC 内的工作负载与其互联网网关之间的流量路由。借助此功能，VPC Route Server 使用首选的 IPv4 或 IPv6 路由动态更新 VPC 和互联网网关路由表，以实现对这些工作负载的路由容错能力。这使您能够自动重新路由 VPC 内的流量，从而提高 VPC 路由的可管理性，以及与第三方工作负载的互操作性。

路由服务器支持以下路由表类型：

- 未与子网关联的 VPC 路由表
- 子网路由表
- 互联网网关路由表

路由服务器不支持与虚拟专用网关关联的路由表。要将路由传播到中转网关路由表，请使用 Transit Gateway Connect。

配额

有关与 Amazon VPC Route Server 相关的配额，请参阅路由服务器配额。

定价

有关与 Amazon VPC Route Server 相关费用的信息，请参阅 Amazon VPC 定价页面上的 VPC 路由服务器选项卡。

内容

- 术语
- Amazon VPC Route Server 的工作原理
- 路由服务器对等日志记录
- 入门教程

术语

本指南中使用了以下术语：

- FIB：转发信息库（FIB）用作路由服务器在评估所有可用路由信息和策略后确定为 RIB 中最佳路径路由的转发表。安装在路由表上的 FIB 路由。每当 RIB 发生变化时，就会重新计算 FIB。
- RIB：路由信息库（RIB）是一个数据库，用于存储路由器或路由系统收集的所有路由信息和网络拓扑数据，例如从 BGP 对等获取的路由。当收到新的路由信息或现有路由发生变化时，RIB 会不断更新。这样可以确保路由服务器始终拥有最新的网络拓扑视图，并能够做出最佳的路由决策。
- 路由服务器：路由服务器组件使用转发信息库（FIB）中的 IPv4 或 IPv6 路由更新您的 VPC 和互联网网关路由表。路由服务器代表单个 FIB 和路由信息库（RIB）。
- 路由服务器关联：路由服务器关联是在路由服务器与 VPC 之间建立的连接。
- 路由服务器端点：路由服务器端点是子网内的 AWS 托管组件，可促进路由服务器与 BGP 对等之间的 BGP（边界网关协议）连接。
- 路由服务器对等：路由服务器对等是路由服务器端点与部署在 AWS 中的设备（例如在 EC2 实例上运行的防火墙设备或其他网络安全功能）之间的会话。设备必须满足以下要求：
  - 在 VPC 中有弹性网络接口
  - 支持 BGP（边界网关协议）
  - 可以启动 BGP 会话
- 路由服务器传播：启用后，路由服务器传播会将路由安装到指定路由表的 FIB 中。路由服务器支持 IPv4 和 IPv6 路由传播。

Amazon VPC Route Server 的工作原理

本节说明 Amazon VPC Route Server 的工作原理，并帮助您了解其如何为子网中运行的工作负载实现路由容错能力。

内容

- 概览
- 示意图

概览

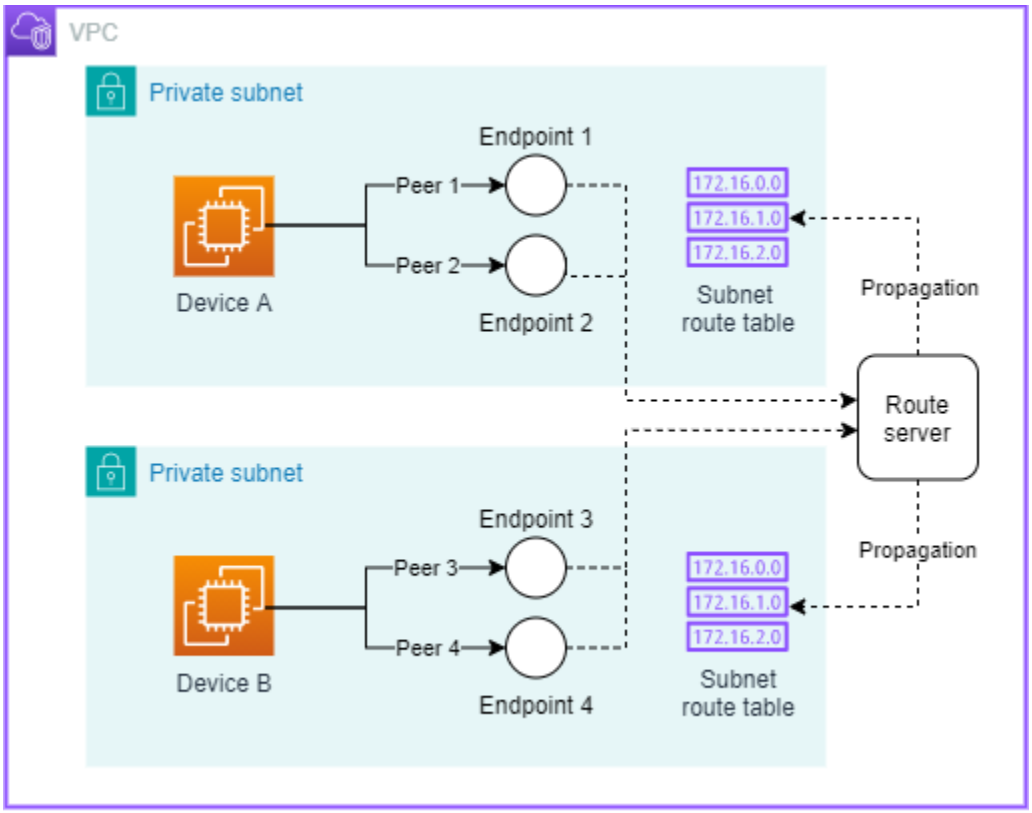
Amazon VPC Route Server 的工作原理：

1. 您可以将网络设备（例如在 VPC 中 EC2 实例上运行的防火墙）配置为使用 Amazon VPC Route Server。
2. 网络设备出现故障。

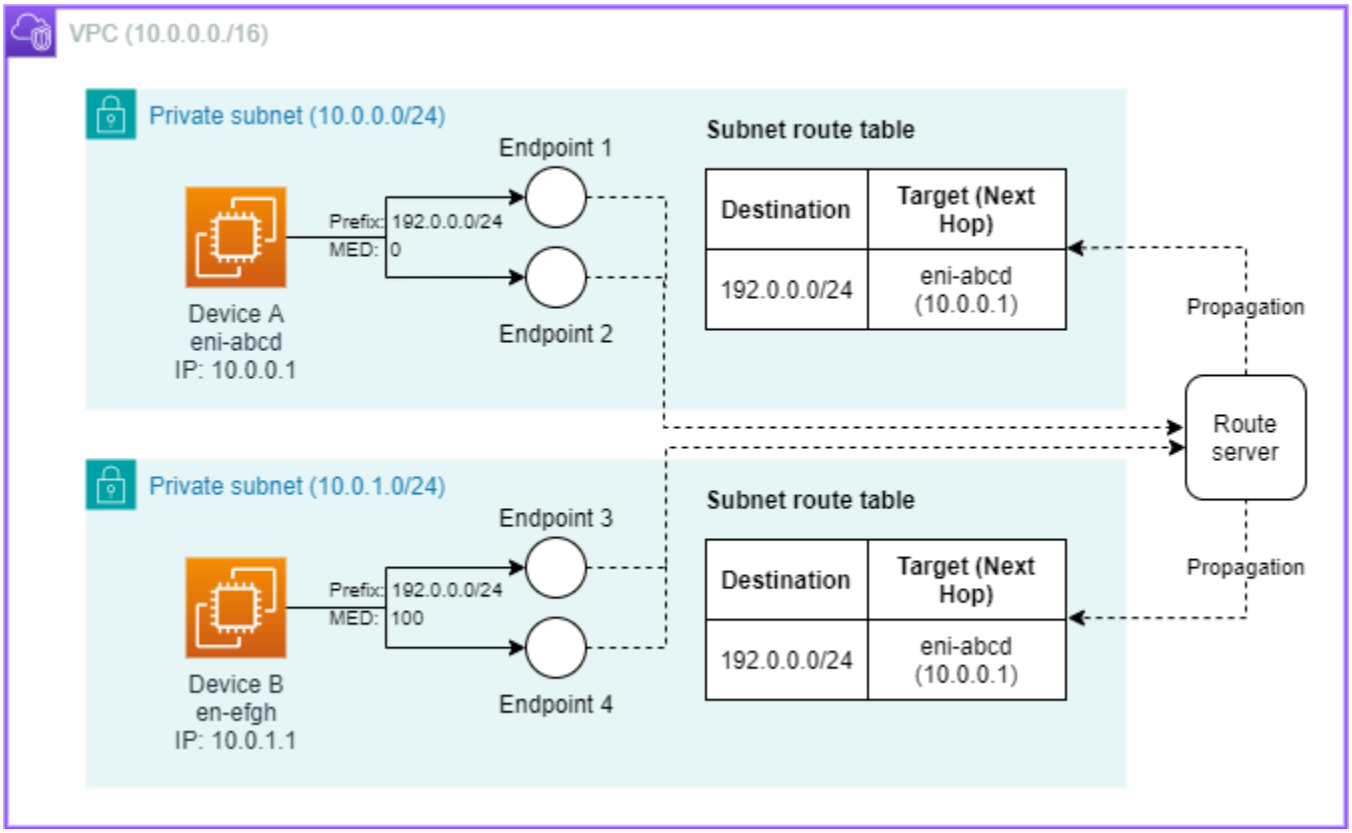
3. 路由服务器端点通过在路由服务器对等上配置的 BFD（双向转发检测）来检测故障。
4. 路由服务器端点会更新路由服务器，以从故障设备所在的下一个跃点的跳路由信息库（RIB）中撤回路由。
5. 路由服务器从 RIB 计算转发信息库（FIB），选择最佳可用路由。
6. 路由服务器使用来自 FIB 的路由更新配置的路由表。
7. 所有新流量都将转发到备用设备。

示意图

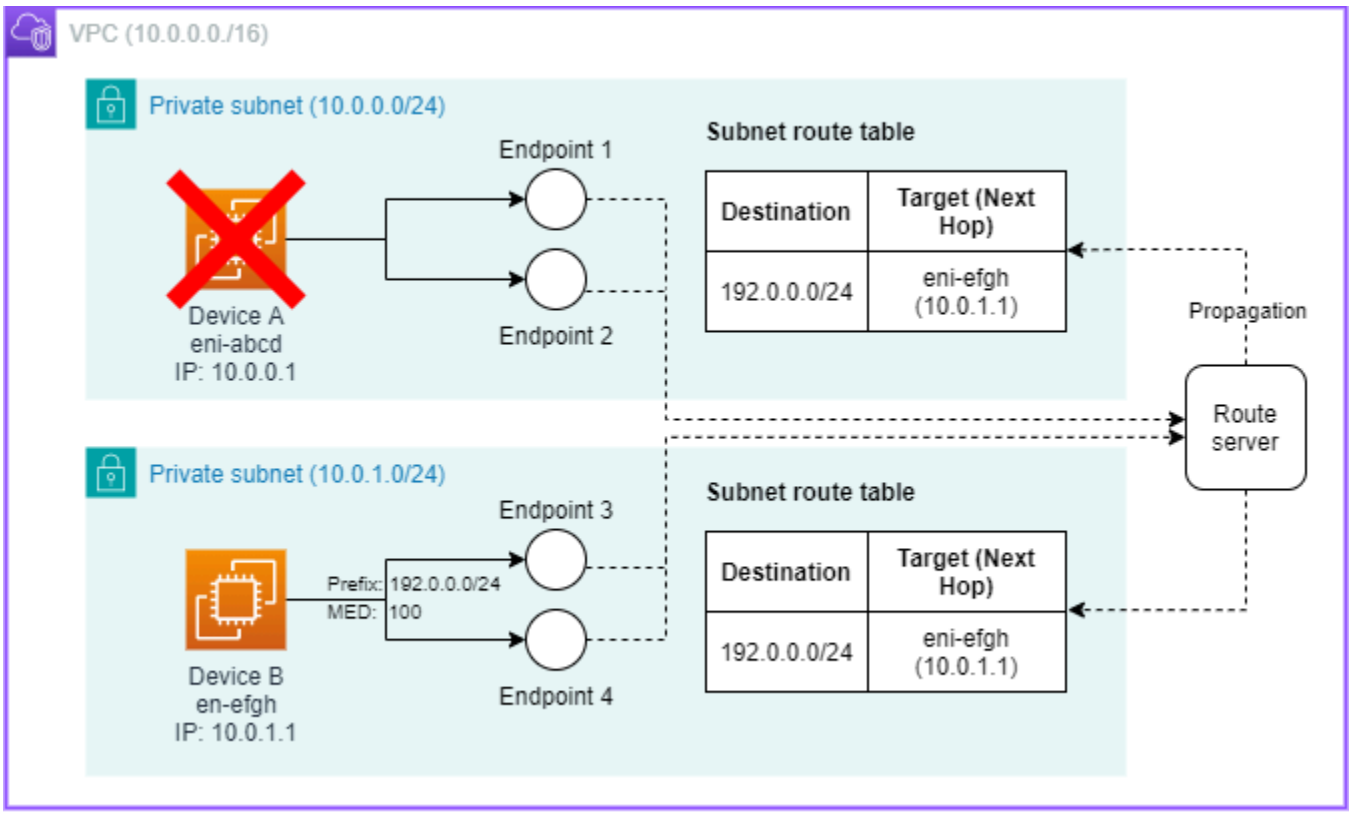
以下是为两个子网中的设备配置路由服务器端点的 VPC 路由服务器示例图。



以上面的示例为基准，下面的示例显示了一个更详细的设计，其中设备 A 和设备 B 都通过 BGP 通告其可以接受目标 IP 在 192.0.0.0/24 范围内（从 192.0.0.0 到 192.0.0.255 ）的任何流量。MED（多出口标识）属性为 0 告诉路由服务器，设备 A 应该优先于设备 B。路由服务器从设备 A 接收路由和 MED 属性，并将该路由安装到子网路由表中，将设备 A 的网络接口作为“下一个跃点”。因此，子网内目标 IP 在 192.0.0.0/24 范围内的任何流量都将发送到设备 A。然后设备 A 将处理该流量并将其继续发送。任一子网（10.0.0.0/24 或 10.0.1.0/24）内发往 192.0.0.0/24 的流量将路由到设备 A eni-abcd（10.0.0.1），作为下一个跃点。



下面的最后一个示例显示了路由服务器如何处理失效转移。虽然较高的 MED 属性告诉路由服务器设备 B 的优先级低于设备 A，但如果设备 A eni-abcd（10.0.0.1）发生故障，则路由服务器会更新子网路由表，并将发往 192.0.0.0/24 的流量路由到设备 B eni-efgh（10.0.1.1）作为下一个跃点。



路由服务器对等日志记录

当需要执行以下操作时，请使用 VPC 路由服务器对等日志记录：

- 监控 BGP 和 BFD 会话运行状况
- 排查连接问题
- 查看历史会话更改
- 跟踪网络状态

定价

- CloudWatch：当您将路由服务器对等日志发布到 CloudWatch Logs 时，将收取已出售日志的数据摄取和存档费用。
- S3：当您将路由服务器对等日志发布到 Amazon S3 时，将收取已出售日志的数据摄取和存档费用。
- Data Firehose：收取标准摄取和传输费用。

已出售日志是来自特定 AWS 服务的日志，这些服务按批量分层定价提供，并传送到 CloudWatch Logs、Amazon S3 或 Amazon Data Firehose。有关更多信息，请打开 Amazon CloudWatch Pricing（Amazon CloudWatch 定价），选择 Logs（日志），找到 Vended Logs（已出售日志）。

日志格式示例

```
{
    "resource_arn": "arn:aws:ec2:us-east-1:111122223333:route-server-peer/
rsp-1234567890abcdef0",
    "event_timestamp": 1746643505367,
    "type": "RouteStatus",
    "status": "ADVERTISED",
    "message": {
        "prefix": "10.24.34.0/32",
        "asPath": "65000",
        "med": 100,
        "nextHopIp": "10.24.34.1"
    }
}

{
    "resource_arn": "arn:aws:ec2:us-east-1:111122223333:route-server-peer/
rsp-1234567890abcdef0",
    "event_timestamp": 1746643490000,
    "type": "BGPStatus",
    "status": "UP",
    "message": null
}
```

其中：

- resource_arn 是路由服务器对等体的 ARN。
- event_timestamp 是事件的时间戳。
- 我们生成的日志事件的 type（RouteStatus、BGPStatus、BFDStatus）。
- status 字段是状态更新。
  - 对于 RouteStatus 类型消息
    - ADVERTISED（路由由对等体公布）
    - UPDATED（现有路由由对等体更新）
    - WITHDRAWN（路由由对等体撤回）

  - 对于 BFDStatus 和 BGPStatus 更新
    - UP, DOWN.
- message 字段当前仅用于 RouteStatus 消息类型的路由属性，但可以填充任何类型的相关信息。

AWS Management Console

要创建路由服务器对等日志：

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中的虚拟私有云下，选择路由服务器。
3. 在路由服务器页面上，选择路由服务器对等。
4. 选择日志传输选项卡。
5. 选择添加日志传输。
6. 选择一个目的地并配置设置：

   - Amazon CloudWatch Logs
     - 日志类型：要传输的日志类型。唯一支持的日志类型是 EVENT_LOGS。
     - 目的地日志组：将发送日志的 CloudWatch 日志组。您可以选择一个现有日志组或创建一个新的日志组（例如：/aws/vpc/route-server-peers）。
     - 字段选择：要包括在日志中的数据字段。
     - 输出格式：如何设置日志的格式：
       - JSON：计算机处理的结构化格式
       - 文本：纯文本格式
     - 字段分隔符：使用文本格式时，这是用于分隔字段的字符（例如：逗号、制表符、空格）。
   - Amazon S3
     - 跨账户 - 向不同的 AWS 账户发送日志
       - 日志类型：要传输的日志类型。唯一支持的日志类型是 EVENT_LOGS。
       - 传输目的地 ARN：将发送日志的另一个 AWS 账户中 S3 存储桶的 Amazon 资源名称。
       - 字段选择：要包括在日志中的数据字段。
       - 后缀：添加到日志文件名的结尾（例如：.log、.txt）。
       - Hive 兼容：启用后，将日志组织到可与基于 Hive 的工具配合使用的文件夹结构中，以便更轻松地使用 Amazon Athena 等服务进行搜索。

- 字段分隔符：使用文本格式时，这是用于分隔字段的字符。
- 在当前账户中
  - 日志类型：要传输的日志类型。唯一支持的日志类型是 EVENT_LOGS。
  - 目的地 S3 存储桶：您的账户中将发送日志的 S3 存储桶。您可以指定一个子文件夹路径。
  - 字段选择：要包括在日志中的数据字段。
  - 后缀：添加到日志文件名的结尾（例如：.log、.txt）。
  - Hive 兼容：启用后，将日志组织到可与基于 Hive 的工具配合使用的文件夹结构中，以便更轻松地进行搜索。
  - 字段分隔符：使用文本格式时，这是用于分隔字段的字符。
- Amazon Data Firehose
  - 跨账户
    - 日志类型：要传输的日志类型。唯一支持的日志类型是 EVENT_LOGS。
    - 传输目的地 ARN：另一个 AWS 账户中的 Firehose 传输流的 Amazon 资源名称。
    - 字段选择：要包括在日志中的数据字段。
    - 字段分隔符：使用文本格式时，这是用于分隔字段的字符。
  - 在当前账户中
    - 日志类型：要传输的日志类型。唯一支持的日志类型是 EVENT_LOGS。
    - 传输目的地流：您的账户中将发送日志的 Firehose 传输流。流必须使用“Direct Put”源类型。
    - 字段选择：要包括在日志中的数据字段。
    - 输出格式：如何设置日志的格式：
      - JSON：计算机处理的结构化格式
      - 文本：纯文本格式
    - 字段分隔符：使用文本格式时，这是用于分隔字段的字符。

Command line

本部分的命令链接到 AWS CLI 参考文档。本文档提供了运行命令时可以使用的选项的详细说明。

要创建路由服务器对等日志：

1. 使用 put-delivery-source 命令。

- 示例请求

```
aws logs put-delivery-source --name "source-rsp-1234567890abcdef0" --
resource-arn "arn:aws:ec2:us-east-1:111122223333:route-server-peer/
rsp-1234567890abcdef0" --log-type "EVENT_LOGS"
```

- 响应示例

```
{
     "deliverySource": {
        "name": "source-rsp-1234567890abcdef0",
        "arn": "arn:aws:logs:us-east-1:111122223333:delivery-source:source-
rsp-1234567890abcdef0",
        "resourceArns": [
            "arn:aws:ec2:us-east-1:111122223333:route-server-peer/
rsp-1234567890abcdef0"
        ],
        "service": "ec2",
        "logType": "EVENT_LOGS"
    }
}
```

2. 使用 put-delivery-destination 命令。

- 以下 AWS CLI 示例创建路由服务器日志。日志将传输到指定的日志组。
- 示例请求

```
aws logs put-delivery-destination --name "destination-rsp-abcdef01234567890"
 --destination-resource-arn "arn:aws:logs:us-east-1:111122223333:log-group:/
aws/vendedlogs/ec2/route-server-peer/EVENT_LOGS/rsp-abcdef01234567890"
```

- 响应示例

```
{
     "deliveryDestination": {
        "name": "destination-rsp-abcdef01234567890",
        "arn": "arn:aws:logs:us-east-1:111122223333:delivery-
destination:destination-rsp-abcdef01234567890",
        "deliveryDestinationType": "CWL",
        "deliveryDestinationConfiguration": {
```

```
            "destinationResourceArn": "arn:aws:logs:us-
east-1:111122223333:log-group:/aws/vendedlogs/ec2/route-server-peer/
EVENT_LOGS/rsp-abcdef01234567890"
        }
    }
}
```

3. 使用 create-delivery 命令。

   - 示例请求

```
aws logs create-delivery --delivery-source-name "source-rsp-1234567890abcdef0"
 --delivery-destination-arn "arn:aws:logs:us-east-1:111122223333:delivery-
destination:destination-rsp-abcdef01234567890"
```

   - 响应示例

```
{
     "delivery": {
        "id": "1234567890abcdef0",
        "arn": "arn:aws:logs:us-
east-1:111122223333:delivery:1234567890abcdef0",
        "deliverySourceName": "source-rsp-1234567890abcdef0",
        "deliveryDestinationArn": "arn:aws:logs:us-
east-1:111122223333:delivery-destination:destination-rsp-abcdef01234567890",
        "deliveryDestinationType": "CWL",
        "recordFields": [
            "resource_arn",
            "event_timestamp",
            "type",
            "status",
            "message"
        ]
    }
}
```

入门教程

本教程将指导您完成设置和配置 VPC Route Server，以在 VPC 中启用动态路由的过程。您将学习如何创建和配置所有必要的组件、建立 BGP 对等，以及验证操作是否正确。本教程涵盖了从初始 IAM 设置到测试和清理的所有内容。

开始本教程之前，请确保您满足以下条件：

- AWS 账户的管理权限
- 具有至少两个要在其中启用动态路由的子网的 VPC
- 支持 BGP 并可用作路由服务器对等设备的网络设备（如在 EC2 实例上运行的防火墙）
- 基本熟悉 BGP 概念和 AWS 网络

这些步骤可以使用 AWS 管理控制台或 AWS CLI 完成。每个步骤都提供了两种方法。

预计完成时间：15 到 30 分钟

步数

- 步骤 1：配置所需的 IAM 角色权限
- 步骤 2：创建路由服务器
- 步骤 3：将路由服务器与 VPC 关联
- 步骤 4：创建路由服务器端点
- 步骤 5：启用路由服务器传播
- 步骤 6：创建路由服务器对等
- 步骤 7：从设备启动 BGP 会话
- 步骤 8：清除

步骤 1：配置所需的 IAM 角色权限

要使用 VPC Route Server，请确保您使用的 IAM 用户或角色具有所需的 IAM 权限。以下是每个 API 需要哪些权限的指南：

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CreateRouteServer",
            "Effect": "Allow",
            "Action": [
                "sns:CreateTopic"
```

```
            ],
            "Resource": "*"
        },
        {
            "Sid": "DeleteRouteServer",
            "Effect": "Allow",
            "Action": [
                "sns:DeleteTopic"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CreateRouteServerEndpoint",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:CreateSecurityGroup",
                "ec2:DescribeSecurityGroups",
                "ec2:AuthorizeSecurityGroupIngress",
                "ec2:CreateTags",
                "ec2:DeleteTags"
            ],
            "Resource": "*"
        },
        {
            "Sid": "DeleteRouteServerEndpoint",
            "Effect": "Allow",
            "Action": [
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteSecurityGroup",
                "ec2:RevokeSecurityGroupIngress",
                "ec2:CreateTags",
                "ec2:DeleteTags"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CreateRouteServerPeer",
            "Effect": "Allow",
            "Action": [
                "ec2:AuthorizeSecurityGroupIngress"
            ],
            "Resource": "*"
```

```
        },
        {
            "Sid": "DeleteRouteServerPeer",
            "Effect": "Allow",
            "Action": [
                "ec2:RevokeSecurityGroupIngress"
            ],
            "Resource": "*"
        }
    ]
}
```

步骤 2：创建路由服务器

完成本部分中的步骤以创建路由服务器。

路由服务器组件使用转发信息库（FIB）中的 IPv4 或 IPv6 路由更新您的 VPC 和互联网网关路由表。路由服务器代表单个 FIB 和路由信息库（RIB）。

AWS Management Console

创建路由服务器

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中的虚拟私有云下，选择路由服务器。
3. 在路由服务器页面上，选择创建路由服务器。
4. 在创建路由服务器页面上，配置以下设置：
   - 对于名称，输入路由服务器的名称（例如，"my-route-server-01"）。名称长度不得超过 255 个字符。
   - 对于 Amazon 端 ASN，输入 BGP ASN 值。该值必须在 1 到 4294967295 的范围内。建议使用 64512 到 65534（16 位 ASN）或 4200000000 到 4294967294（32 位 ASN）范围内的专用 ASN。
   - 对于保留路由，请选择启用或禁用。此选项决定在所有 BGP 会话终止后是否应保留路由：
     - 如果启用：即使所有 BGP 会话均已结束，路由仍将保留在路由服务器的路由数据库中
     - 如果禁用：所有 BGP 会话结束后，路由将从路由数据库中删除

- 如果您启用了保留路由，请在保留持续时间中输入一个介于 1 到 5 分钟之间的值。此持续时间指定路由服务器在重新建立 BGP 后要等待多长时间才能取消保留路由。例如，如果您将其设置为 1 分钟，则在重新建立 BGP 后，您的设备有 1 分钟的时间来重新学习和通告其路由，然后路由服务器将恢复正常功能。虽然 1 分钟通常就足够了，但如果您的 BGP 网络需要更多时间来完全重新建立和重新学习所有路由，则可以设置最多 5 分钟。
- （可选）要启用 BGP 状态更改的 SNS 通知，请切换启用 SNS 通知开关。启用 SNS 通知功能后，路由服务器对等上的 BGP 或 BFD 会话状态更改，以及路由服务器端点的维护通知，都将保留到 AWS 预置的 SNS 主题。有关这些通知的详细信息，请参阅下方的 SNS 通知详细信息表。

5. （可选）要向您的路由服务器添加标签，请向下滚动到标签 - 可选部分，然后选择添加新标签。输入每个标签的键和可选值。最多可以添加 50 个标签。
6. 检查您的设置，然后选择创建路由服务器。
7. 等待创建路由服务器。完成后，您将重定向到路由服务器页面，在此可以看到列出的新路由服务器，其状态为可用。

Command line

使用以下步骤创建新的路由服务器，以管理 VPC 中的动态路由。

对于 `--amazon-side-asn`，输入 BGP ASN 值。该值必须在 1 到 4294967295 的范围内。建议使用 64512 到 65534（16 位 ASN）或 4200000000 到 4294967294（32 位 ASN）范围内的专用 ASN。

1. 命令:

```
aws ec2 create-route-server --amazon-side-asn 65000
```

响应：

```
{
    "RouteServer": {
        "RouteServerId": "rs-1",
        "AmazonSideAsn": 65000,
        "State": "pending"
    }
}
```

2. 等待路由服务器变为可用。

命令:

```
aws ec2 describe-route-servers
```

响应：

```
{
    "RouteServer": {
        "RouteServerId": "rs-1",
        "AmazonSideAsn": 65000,
        "State": "available"
    }
}
```

SNS 通知详细信息

下表显示 Amazon VPC Route Server 将使用 Amazon SNS 发送的消息的详细信息：

标准字段		消息属性（元数据）			
消息	何时发送	timestamp	eventCode	routeServerEndpointId	affectedRouteServerPeerIds
路由服务器端点 [ENDPOINT ID] 现在正在进行维护。BFD 和 BGP 会话可能会受到影响。	路由服务器端点维护	格式：2025-02-17T15:55:00Z	ROUTE_SERVER_ENDPOINT_MAINTENANCE	受影响的端点 ID	受影响的对等 ID 列表
消息	何时发送	timestamp	eventCode	routeServerPeerId	newBgpStatus

标准字段		消息属性（元数据）			
路由服务器对等 [PEER ID] 的 BGP 现在处于 [UP/DOWN] 状态。	路由服务器对等 BGP 状态更改	格式：2025-02-17T15:55:00Z	ROUTE_SERVER_PEER_BGP_STATUS_CHANGE	受影响的对等 ID	运行或停机
消息	何时发送	timestamp	eventCode	routeServerPeerId	newBfdStatus
路由服务器对等 [PEER ID] 的 BFD 现在处于 [UP/DOWN] 状态。	路由服务器对等 BFD 状态更改	格式：2025-02-17T15:55:00Z	ROUTE_SERVER_PEER_BFD_STATUS_CHANGE	受影响的对等 ID	运行或停机

步骤 3：将路由服务器与 VPC 关联

完成本部分中的步骤将路由服务器与 VPC 关联。

路由服务器关联是在路由服务器与 VPC 之间建立的连接。这是一个基本配置步骤，使路由服务器能够与 VPC 中的设备协同工作。

创建路由服务器关联时：

- 其将路由服务器关联到特定的 VPC。
- 其使路由服务器能够与 VPC 子网内的路由表进行交互。
- 其允许路由服务器在关联的 VPC 内接收和传播路由。
- 其确定路由服务器可以运行的范围。

路由服务器关联的关键方面：

- 每个路由服务器可以与一个 VPC 关联。默认情况下，每个 VPC 最多可以有 5 个独立的路由服务器关联。有关限额的更多信息，请参阅路由服务器限额。
- 必须先创建关联，然后路由服务器才能管理路由。
- 可以监控关联以追踪其状态（例如正在关联和已关联）。
- 如果您不再希望路由服务器在该 VPC 中运行，则可以移除关联（取消关联）。

AWS Management Console

将路由服务器与 VPC 关联

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中的虚拟私有云下，选择路由服务器。
3. 选择要与 VPC 关联的路由服务器。
4. 在关联选项卡上，选择关联路由服务器。
5. 在“关联路由服务器”对话框中：
   - 路由服务器 ID 字段将自动填充您选择的路由服务器
   - 对于 VPC ID，从下拉列表中选择要关联的 VPC
6. 选择关联路由服务器。
7. 等待关联完成。完成后，该状态将在关联选项卡上显示为已关联。

Command line

使用以下步骤将路由服务器与 VPC 关联。

1. 命令:

```
aws ec2 associate-route-server --route-server-id rs-1 --vpc-id vpc-1
```

响应：

```
{
    "RouteServerAssociation": {
        "RouteServerId": "rs-1",
        "VpcId": "vpc-1",
        "State": "associating"
    }
```

```
}
```

2. 等待关联完成。

   命令:

   ```
   aws ec2 get-route-server-associations --route-server-id rs-1
   ```

   响应：

   ```
   {
       "RouteServerAssociation": {
           "RouteServerId": "rs-1",
           "VpcId": "vpc-1",
           "State": "associated"
       }
   }
   ```

步骤 4：创建路由服务器端点

完成本部分中的步骤，以创建路由服务器端点。为每个子网创建两个端点以实现冗余。

路由服务器端点是子网内的 AWS 托管组件，可促进路由服务器与 BGP 对等之间的 BGP（边界网关协议）连接。

路由服务器端点是网络设备与路由服务器建立 BGP 会话的“接触点”。它们是实际处理 BGP 连接的组件，而路由服务器本身则管理路由决策和路由传播。

> **Note**
>
> 路由服务器端点的费用为每小时 0.75 美元。

AWS Management Console

创建路由服务器端点

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中的虚拟私有云下，选择路由服务器。
3. 选择要为其创建端点的路由服务器。

4. 在下方窗格中，选择路由服务器端点选项卡。
5. 选择创建路由服务器端点。
6. 在创建路由服务器端点页面上，配置以下设置：
   - 对于名称，为您的端点输入一个描述性名称。
   - 对于路由服务器，请确认已选择正确的路由服务器。
   - 对于子网，选择您要在其中创建端点的子网。
7. （可选）要向您的路由服务器端点添加标签，请向下滚动到标签 - 可选部分，然后选择添加新标签。输入每个标签的键和可选值。
8. 检查您的设置，然后选择创建路由服务器端点。
9. 等待创建端点。完成后，您将看到一条成功消息。
10. 重复步骤 5 到 9，使用不同的名称在同一子网中创建第二个端点。
11. 对需要路由服务器端点的每个子网重复步骤 5 到 10。
12. 创建端点后，返回路由服务器的路由服务器端点选项卡。
13. 确认您看到每个子网列出两个端点。
14. 检查每个端点的状态是否为可用。

Command line

使用以下步骤创建路由服务器端点。

1. 命令:

```
aws ec2 create-route-server-endpoint --route-server-id rs-1 --subnet-id subnet-1
```

响应：

```
{
    "RouteServerEndpoint": {
        "RouteServerId": "rs-1",
        "RouteServerEndpointId": "rse-1",
        "VpcId": "vpc-1",
        "SubnetId": "subnet-1",
        "State": "pending"
    }
}
```

2. 创建后可能需要等待几分钟，端点才会变为完全可用。

   命令:

   ```
   aws ec2 describe-route-server-endpoints
   ```

   响应：

   ```
   {
       "RouteServerEndpoint": {
           "RouteServerId": "rs-1",
           "RouteServerEndpointId": "rse-1",
           "VpcId": "vpc-1",
           "SubnetId": "subnet-1",
           "EniId": "eni-123",
           "EniAddress": "10.1.2.3",
           "State": "available"
       }
   }
   ```

   重复上述步骤，在同一子网中使用不同的名称创建第二个端点，为需要路由服务器端点的每个子网创建端点。

步骤 5：启用路由服务器传播

完成此步骤以启用路由服务器传播。

启用后，路由服务器传播会将路由安装到指定路由表的 FIB 中。路由服务器支持 IPv4 和 IPv6 路由传播。

路由服务器传播是自动更新路由表的机制：路由服务器无需手动更新路由表，而是使用来自 FIB 的路由自动将适当的路由传播到已配置的路由表。

路由服务器传播的关键方面：

- 配置
  - 将路由服务器关联到特定的路由表
  - 确定哪些路由表将接收动态路由更新
  - 可以为每个路由表启用或禁用

- 功能
  - 使用从 BGP 对等获知的路由自动更新路由表
  - 根据 BGP 属性传播最佳可用路由
  - 保持指定路由表间的路由一致性
  - 网络条件发生变化时动态更新路由
- 状态
  - 可以启用（路由正在传播）
  - 可以禁用（路由未进行传播）

AWS Management Console

启用路由服务器传播

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 选择要为其启用传播的路由服务器。
3. 在路由服务器详细信息面板中选择传播选项卡。
4. 选择启用传播。
5. 在启用传播对话框中：
   - 路由服务器 ID 将预先填充。
   - 在路由表下，从新传播路由的下拉菜单中选择目标路由表。
6. 选择启用传播进行确认。
7. 等待传播列表中的传播状态更改为“可用”。
8. 验证所选路由表是否显示在传播列表中，状态为可用。

Command line

使用以下步骤启用路由服务器传播。

1. 命令:

```
aws ec2 enable-route-server-propagation --route-table-id rtb-1 --route-server-id
 rs-1
```

响应：

```
{
    "RouteServerRoutePropagation": {
        "RouteServerId": "rs-1",
        "RouteTableId": "rtb-1",
        "State": "pending"
    }
}
```

2. 等待传播状态变为可用。

   命令:

```
aws ec2 get-route-server-propagations --route-server-id rs-1
```

   响应：

```
{
    "RouteServerRoutePropagation": {
        "RouteServerId": "rs-1",
        "RouteTableId": "rtb-1",
        "State": "available"
    }
}
```

步骤 6：创建路由服务器对等

路由服务器对等是路由服务器端点与部署在 AWS 中的设备（例如在 EC2 实例上运行的防火墙设备或其他网络安全功能）之间的会话。设备必须满足以下要求：

- 在 VPC 中有弹性网络接口
- 支持 BGP（边界网关协议）
- 可以启动 BGP 会话

> **Note**
>
> 建议您为每个路由服务器端点创建一个路由服务器对等以实现冗余。

AWS Management Console

创建路由服务器对等

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航路径中，选择 VPC > 路由服务器对等 > 创建路由服务器对等。
3. 在详细信息下，配置以下各项：
   - 名称：输入路由服务器对等的名称（最多 255 个字符）。例如：my-route-server-peer-01
   - 路由服务器端点 ID：从下拉列表中选择路由服务器端点。或者，选择创建路由服务器端点来创建新端点。
   - 对等地址：输入对等的 IPv4 地址。必须是有效的 IP 地址。对等地址必须能够从路由服务器端点访问。
   - 对等 ASN：输入 BGP 对等的 ASN（自治系统编号）。值必须在 1 到 4294967295 的范围内。ASN 通常应使用专用范围（16 位为 64512 到 65534，32 位为 4200000000 到 4294967294）
   - 对等活跃度检测：
     - BGP 保持活跃（默认）：标准 BGP 保持活动状态机制
     - BFD：双向转发检测，可实现更快的失效转移
   - （可选）在标签下，选择添加新标签以添加键值对标签。标签有助于标识和追踪 AWS 资源。
4. 检查您的设置，然后选择创建路由服务器对等。

Command line

使用以下步骤创建路由服务器对等。

1. 命令:

```
aws ec2 create-route-server-peer --route-server-endpoint-id rse-1 --peer-address
 10.0.2.3 --bgp-options PeerAsn=65001,PeerLivenessDetection=bfd
```

响应：

在响应中，状态值可以是 `pending|available|deleting|deleted`。

```
{
```

```
    "RouteServerPeer": {
        "RouteServerPeerId": "rsp-1",
        "RouteServerId": "rs-1",
        "VpcId": "vpc-1",
        "SubnetId": "subnet-1",
        "State": "pending",
        "EndpointEniId": "eni-2,
        "EndpointEniAddress": "10.0.2.4",
        "PeerEniId": "eni-1",
        "PeerAddress": "10.0.2.3",
        "BgpOptions": {
            "PeerAsn": 65001,
   "PeerLivenessDetection": "bfd"
        },
        "BgpStatus": {
            "Status": "Up"
        }
    }
}
```

2. 等待传播状态变为可用。

命令:

```
aws ec2 describe-route-server-peers
```

响应：

```
{
    "RouteServerPeer": {
        "RouteServerPeerId": "rsp-1",
        "RouteServerId": "rs-1",
        "VpcId": "vpc-1",
        "SubnetId": "subnet-1",
        "State": "available",
        "EndpointEniId": "eni-2,
        "EndpointEniAddress": "10.0.2.4",
        "PeerEniId": "eni-1",
        "PeerAddress": "10.0.2.3",
        "BgpOptions": {
            "PeerAsn": 65001,
   "PeerLivenessDetection": "bfd"
        },
```

```
        "BgpStatus": {
            "Status": "down"
        }
    }
}
```

步骤 7：从设备启动 BGP 会话

当路由服务器对等的状态为可用时，请配置您的工作负载以启动与路由服务器端点的 BGP 会话。

从子网中的设备启动 BGP 会话不在本指南的讨论范围内。路由服务器端点不启动 BGP 会话。

您可以通过验证路由表中是否包含路由服务器传播的最佳路由，来检查 VPC Route Server 功能是否正常运行。

步骤 8：清除

本教程的构建部分已完成。完成本部分中的步骤，以删除您创建的 VPC Route Server 组件。

7.1：撤销设备上的 BGP 通告

撤销子网中设备上的 BGP 通告不在本指南的讨论范围内。如有必要，请向第三方供应商咨询您的 BGP 配置。

7.2：禁用路由服务器传播

使用以下步骤禁用路由服务器传播。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 选择要为其禁用传播的路由服务器。
3. 选择操作 > 修改路由服务器。
4. 在路由服务器详细信息面板中选择传播选项卡。
5. 选择要禁用的传播，然后选择禁用传播。
6. 在对话框中，选择禁用路由服务器传播。

Command line

1. 禁用传播：

```
aws ec2 disable-route-server-route-propagation --route-table-id rtb-1 --route-
server-id rs-1
```

2. 确认传播已删除：

```
aws ec2 get-route-server-route-propagations --route-server-id rs-1 [--route-
table-id rtb-1]
```

7.3：删除路由服务器对等

使用以下步骤删除路由服务器对等。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航路径中，选择路由服务器 > 路由服务器对等。
3. 选择路由服务器对等。
4. 选择操作 > 删除路由服务器对等。

Command line

1. 删除对等：

```
aws ec2 delete-route-server-peer --route-server-peer-id rsp-1
```

2. 确认删除操作：

```
aws ec2 describe-route-server-peers [--route-server-peer-ids rsp-1] [--filters
 Key=RouteServerId|RouteServerEndpointId|VpcId]
```

7.4：删除路由服务器端点

使用以下步骤删除路由服务器端点。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 选择要删除端点的路由服务器。
3. 选择路由服务器端点。
4. 选择端点，然后选择操作 > 删除路由服务器端点。
5. 输入删除并选择删除。

Command line

1. 描述端点：

```
aws ec2 describe-route-server-endpoints
```

2. 删除路由服务器端点：

```
aws ec2 delete-route-server-endpoint --route-server-endpoint-id rse-1
```

3. 确认已删除端点：

```
aws ec2 describe-route-server-endpoints [--route-server-endpoint-ids rsp-1] [--
filters Key=RouteServerId|VpcId|SubnetId]
```

7.5：取消路由服务器与 VPC 的关联

使用以下步骤取消路由服务器与 VPC 的关联。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 选择要取消关联的路由服务器。
3. 选择关联。
4. 选择取消关联路由服务器。
5. 确认将要进行的更改，然后选择取消关联路由服务器。

Command line

1. 取消路由服务器与 VPC 的关联：

```
aws ec2 disassociate-route-server --route-server-id rs-1 --vpc-id vpc-1
```

2. 确认取消关联：

```
aws ec2 get-route-server-associations --route-server-id rs-1
```

7.6 删除路由服务器

使用以下步骤删除路由服务器。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 选择要删除的路由服务器。
3. 选择操作 > 删除路由服务器。
4. 输入删除并选择删除。

Command line

1. 删除路由服务器：

```
aws ec2 delete-route-server --route-server-id rs-1
```

2. 确认删除操作：

```
aws ec2 describe-route-servers [--route-server-ids rs-1] [--filters Key=VpcId]
```

Amazon VPC Route Server 教程已完成。

排查 VPC 中的可达性问题

Reachability Analyzer 是一款静态配置分析工具。使用 Reachability Analyzer 可分析和调试 VPC 中两个资源之间的网络可达性。如果可以访问这些资源，则 Reachability Analyzer 会生成有关这些资源间虚拟路径的逐跳详细信息，否则会确定障碍组件。例如，其可以识别缺失或配置错误的路由表路由。

有关更多信息，请参阅 Reachability Analyzer 角色指南。

中间盒路由向导

如果要配置对进入或离开 VPC 的流量路由路径的精细控制（例如，通过将流量重新导向到安全设备），则可以使用 VPC 控制台中的中间盒路由向导。中间盒路由向导通过自动创建必要的路由表和路由（跃点）来帮助您根据需要重新导向流量。

中间盒路由向导可以帮助您针对以下场景配置路由：

- 将流量路由到中间盒设备，例如，配置为安全设备的 Amazon EC2 实例。
- 将流量路由到网关负载均衡器端点。有关更多信息，请参阅网关负载均衡器用户指南。

有关更多信息，请参阅 the section called “中间盒场景”。

内容

- 中间盒路由向导先决条件
- 将 VPC 流量重定向到安全设备
- 中间盒路由向导注意事项
- 中间盒场景

中间盒路由向导先决条件

审核the section called “中间盒路由向导注意事项”。然后，请确保您在使用中间盒路由向导之前具有以下信息。

- VPC。
- 流量进出 VPC 的资源，例如，互联网网关、虚拟私有网关或网络接口。
- 中间盒网络接口或网关负载均衡器端点。
- 流量的目的地子网。

将 VPC 流量重定向到安全设备

中间盒路由向导位于 Amazon Virtual Private Cloud Console 中。

内容

- 1. 使用中间盒路由向导创建路由
- 2. 修改中间盒路由
- 3. 删除中间盒路由向导配置

1. 使用中间盒路由向导创建路由

使用中间盒路由向导创建路由

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Your VPCs(您的 VPC)。
3. 选择您的 VPC，然后选择 Actions (操作)、Manage middlebox routes (管理中间盒路由)。
4. 选择 Create routes (创建路由)。
5. 在 Specify routes (指定路由) 页面中，执行以下操作：
   - 对于 Source (源)，选择流量的来源。如果选择虚拟私有网关，则对于 Destination IPv4 CIDR (目的地 IPv4 CIDR)，输入从虚拟私有网关进入 VPC 的本地流量的 CIDR。
   - 对于 Middlebox (中间盒)，选择与您的中间盒设备关联的网络接口 ID，或者当您使用网关负载均衡器端点时，选择 VPC 终端节点 ID。
   - 对于 Destination subnet (目的地子网)，选择目的地子网。
6. （可选）要添加其他目的地子网，请选择 Add additional subnet (添加其他子网)，然后执行以下操作：
   - 对于 Middlebox (中间盒)，选择与您的中间盒设备关联的网络接口 ID，或者当您使用网关负载均衡器端点时，选择 VPC 终端节点 ID。

     对于多个子网，您必须使用同一个中间盒设备。
   - 对于 Destination subnet (目的地子网)，选择目的地子网。
7. （可选）要添加其他源，请选择 Add source (添加源)，然后重复前面的步骤。
8. 选择下一步。
9. 在 Review and create (审核和创建) 页面上，验证路由，然后选择 Create routes (创建路由)。

2. 修改中间盒路由

您可以通过更改网关、中间盒或目的地子网来编辑路由配置。

进行任何修改时，中间盒路由向导会自动执行以下操作：

- 为网关、中间盒和目的地子网创建新路由表。
- 将必要的路由添加到新路由表中。
- 断开中间盒路由向导与资源关联的当前路由表的关联。
- 将中间盒路由向导创建的新路由表与资源相关联。

使用中间盒路由向导修改中间盒路由

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Your VPCs(您的 VPC)。
3. 选择您的 VPC，然后选择 Actions (操作)、Manage middlebox routes (管理中间盒路由)。
4. 选择 Edit routes (编辑路由)。
5. 要更改网关，请在 Source (源) 中，选择流量通过该源进入 VPC 的网关。如果选择虚拟私有网关，则对于 Destination IPv4 CIDR (目的地 IPv4 CIDR)，输入目的地子网 CIDR。
6. 要添加其他目的地子网，请选择 Add additional subnet (添加其他子网)，然后执行以下操作：
   - 对于 Middlebox (中间盒)，选择与您的中间盒设备关联的网络接口 ID，或者当您使用网关负载均衡器端点时，选择 VPC 终端节点 ID。

     对于多个子网，您必须使用同一个中间盒设备。
   - 对于 Destination subnet (目的地子网)，选择目的地子网。
7. 选择下一步。
8. 在 Review and update (审核和更新) 页面上，显示将由中间盒路由向导创建的路由表及其路由的列表。验证路由，然后在确认对话框中，选择 Update routes (更新路由)。

3. 删除中间盒路由向导配置

如果您决定不再需要中间盒路由向导配置，则必须手动删除路由表。

删除中间盒路由向导配置

1. 查看中间盒路由向导路由表。

执行此操作后，中间盒路由向导创建的路由表将显示在单独的路由表页上。

2. 删除显示的每个路由表。

中间盒路由向导注意事项

在使用中间想路由向导时，请注意以下事项：

- 如果要检查流量，则可以为源使用互联网网关或虚拟私有网关。
- 如果您在同一 VPC 中的多个中间盒配置中使用相同的中间盒，请确保两个子网的中间盒处于同一跃点位置。
- 设备必须在源流量或目的地子网的单独子网中配置。
- 您必须禁用设备上的源/目标检查。有关更多信息，请参阅《Amazon EC2 用户指南》中的更改源或目标检查。
- 中间盒路由向导创建的路由表和路由计入您的配额。有关更多信息，请参阅 the section called “路由表”。
- 如果删除资源（例如网络接口），则路由表与资源的关联将被删除。如果资源是目标，则路由目的地设置为黑洞。路由表不会被删除。
- 中间盒子网和目的地子网必须与非默认路由表关联。

> **Note**
>
> 我们建议您使用中间盒路由向导修改或删除使用中间盒路由向导创建的任何路由表。

- 如果使用中间盒路由通过安全设备进行路由，则不支持检查后在源和最终目的地之间进行安全组引用。

中间盒场景

Amazon Virtual Private Cloud（VPC）提供广泛的联网功能，有助于您自定义和控制虚拟网络中的流量路由。中间盒路由向导就是这样一种功能，可以精细控制进出 VPC 的流量的路由路径。

如需将流量重定向到安全设备、负载均衡器或其他网络设备，以便进行检查、监控或优化，中间盒路由向导可以简化流程。此向导可自动创建必要的路由表和路由（跃点），帮助您根据需要重定向指定的流量，免于手动设置复杂的路由配置。

中间盒路由向导支持几种不同的使用场景。例如，可以使用该向导来检查发往特定子网的流量、在整个 VPC 中配置中间盒流量路由和检查，或者有选择地检查特定子网之间的流量。这种对流量路由的精细控制有助于您实施高级安全策略、启用集中式网络监控或优化基于云的应用程序的性能。

以下示例描述了中间盒路由向导的场景。

内容

- 检查发往子网的流量
- 在 VPC 中配置中间盒流量路由和检查
- 检查子网之间的流量

检查发往子网的流量

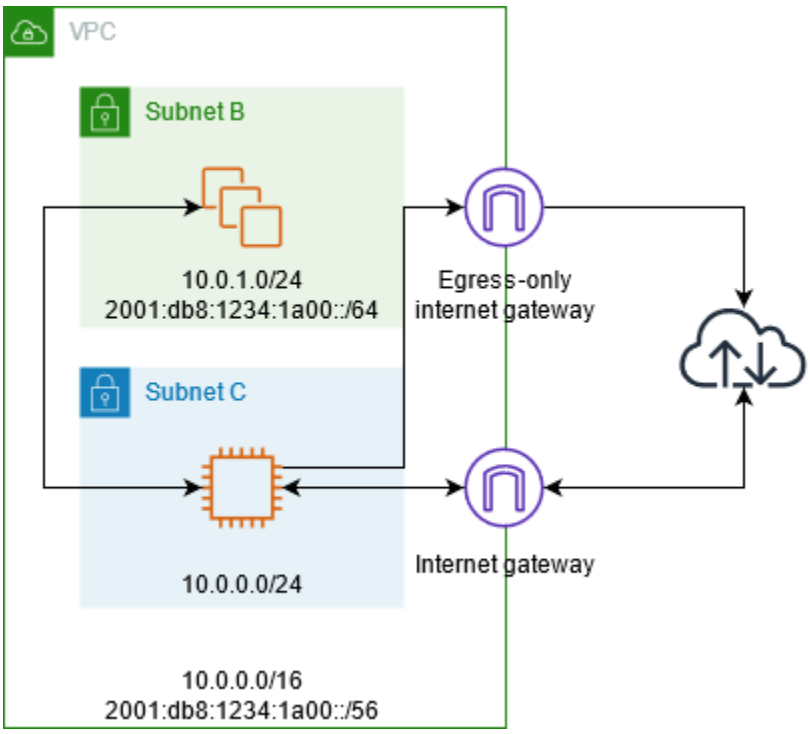
请考虑一下，您的流量通过互联网网关进入 VPC，并且希望使用 EC2 实例上安装的防火墙设备检查发往子网（例如子网 B）的所有流量。防火墙设备应安装和配置在与 VPC 中子网 B 不同的子网（例如子网 C）中的 EC2 实例上，然后您可以使用中间盒路由向导为子网 B 和互联网网关之间的流量配置路由。

中间盒路由向导会自动执行以下操作：

- 创建以下路由表：
  - 互联网网关的路由表
  - 目标子网的路由表
  - 中间盒子网的路由表
- 将必要的路由添加到新路由表中，如以下部分所述。
- 取消与互联网网关、子网 B 和子网 C 关联的当前路由表的关联。
- 将路由表 A 与互联网网关（中间盒路由向导中的 Source（源））相关联、路由表 C 与子网 C（中间盒路由向导中的 Middlebox（中间盒））相关联，并将路由表 B 与子网 B（中间盒路由向导中的 Destination（目的地））相关联。
- 创建一个标签，指示它是由中间盒路由向导创建的，并创建一个指示创建日期的标签。

中间盒路由向导不会修改现有的路由表。它会创建新的路由表，然后将它们与您的网关和子网资源相关联。如果您的资源已与现有路由表显式关联，则首先取消现有路由表的关联，然后将新路由表与您的资源相关联。您的现有路由表不会被删除。

如果不使用中间盒路由向导，则必须手动配置路由表，然后将路由表分配给子网和互联网网关。



互联网网关路由表

将以下路由添加到互联网网关的路由表中。

目标位置	目标	用途
10.0.0.0/16	本地	IPv4 的本地路由
10.0.1.0/24	*appliance-eni*	将发往子网 B 的 IPv4 流量路由到中间盒
2001:db8:1234:1a00::/56	本地	IPv6 的本地路由
2001:db8:1234:1a00::/64	*appliance-eni*	将发往子网 B 的 IPv6 流量路由到中间盒

互联网网关和 VPC 之间存在边缘关联。

使用中间盒路由向导时，它将以下标签与路由表相关联：

- 键为“Origin”，值为“Middlebox wizard”
- 键为“date_created”，值为创建时间（例如“2021-02-18T22:25:49.137Z”）

目标子网路由表

将以下路由添加到目标子网（示例图中的子网 B）的路由表中。

目标位置	目标	用途
10.0.0.0/16	本地	IPv4 的本地路由
0.0.0.0/0	*appliance-eni*	将发往互联网的 IPv4 流量路由到中间盒
2001:db8:1234:1a00::/56	本地	IPv6 的本地路由
::/0	*appliance-eni*	将发往互联网的 IPv6 流量路由到中间盒

与中间盒子网存在子网关联。

使用中间盒路由向导时，它将以下标签与路由表相关联：

- 键为“Origin”，值为“Middlebox wizard”
- 键为“date_created”，值为创建时间（例如“2021-02-18T22:25:49.137Z”）

中间盒子网路由表

将以下路由添加到中间盒子网（示例图中的子网 C）的路由表中。

目标位置	目标	用途
10.0.0.0/16	本地	IPv4 的本地路由
0.0.0.0/0	*igw-id*	到互联网网关的 IPv4 流量路由
2001:db8:1234:1a00::/56	本地	IPv6 的本地路由
::/0	*eigw-id*	将 IPv6 流量路由到仅出口互联网网关

与目标子网存在子网关联。

使用中间盒路由向导时，它将以下标签与路由表相关联：

- 键为“Origin”，值为“Middlebox wizard”
- 键为“date_created”，值为创建时间（例如“2021-02-18T22:25:49.137Z”）

在 VPC 中配置中间盒流量路由和检查

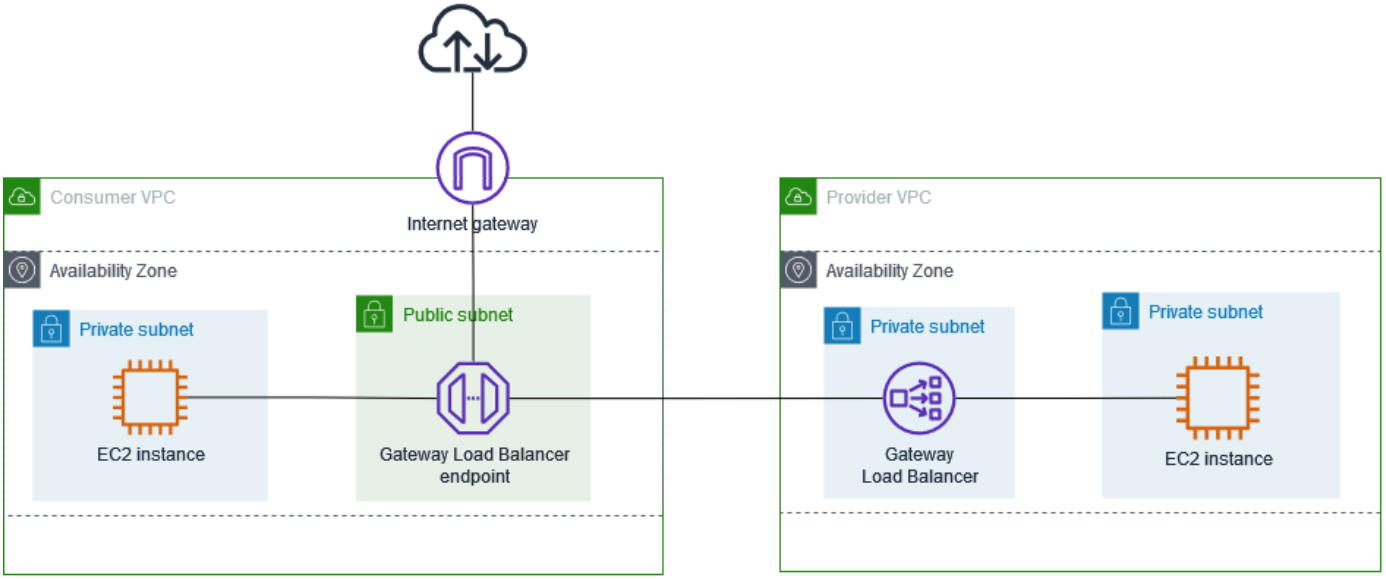
考虑以下场景，其中您需要使用在网关负载均衡器后方配置的安全设备队列检查从互联网网关流入 VPC 并发往子网的流量。服务使用者 VPC 的拥有者在其 VPC 的子网中创建一个网关负载均衡器端点（通过端点网络接口表示）。通过互联网网关进入 VPC 的所有流量首先会路由到网关负载均衡器端点，以便进行检查，然后再路由到应用程序子网。同样，离开应用程序子网中的所有流量首先会路由到网关负载均衡器端点，以便进行检查，然后再路由到互联网。

中间盒路由向导会自动执行以下操作：

- 创建路由表。
- 将必要的路由添加到新路由表中。
- 取消与子网关联的当前路由表的关联。
- 将中间盒路由向导创建的路由表与子网相关联。
- 创建一个标签，指示它是由中间盒路由向导创建的，并创建一个指示创建日期的标签。

中间盒路由向导不会修改现有的路由表。它会创建新的路由表，然后将它们与您的网关和子网资源相关联。如果您的资源已与现有路由表显式关联，则首先取消现有路由表的关联，然后将新路由表与您的资源相关联。您的现有路由表不会被删除。

如果不使用中间盒路由向导，则必须手动配置路由表，然后将路由表分配给子网和互联网网关。



互联网网关路由表

互联网网关的路由表有以下路由。

目标位置	目标	用途
### VPC CIDR	本地	本地路由
###### CIDR	*## ID*	将发往应用程序子网的流量路由到网关负载均衡器端点

与网关存在边缘关联。

使用中间盒路由向导时，它将以下标签与路由表相关联：

- 键为“Origin”，值为“Middlebox wizard”
- 键为“date_created”，值为创建时间（例如“2021-02-18T22:25:49.137Z”）

应用程序子网路由表

应用程序子网的路由表具有以下路由：

目标位置	目标	用途
### VPC CIDR	本地	本地路由
0.0.0.0/0	*## ID*	在将流量路由到互联网之前，将流量从应用程序服务器路由到网关负载均衡器端点

使用中间盒路由向导时，它将以下标签与路由表相关联：

- 键为“Origin”，值为“Middlebox wizard”
- 键为“date_created”，值为创建时间（例如“2021-02-18T22:25:49.137Z”）

提供程序子网路由表

提供程序子网的路由表具有以下路由：

目标位置	目标	用途
#### VPC CIDR	本地	本地路由 确保源自互联网的流量路由到应用程序服务器
0.0.0.0/0	*igw-id*	将所有流量路由到互联网网关。

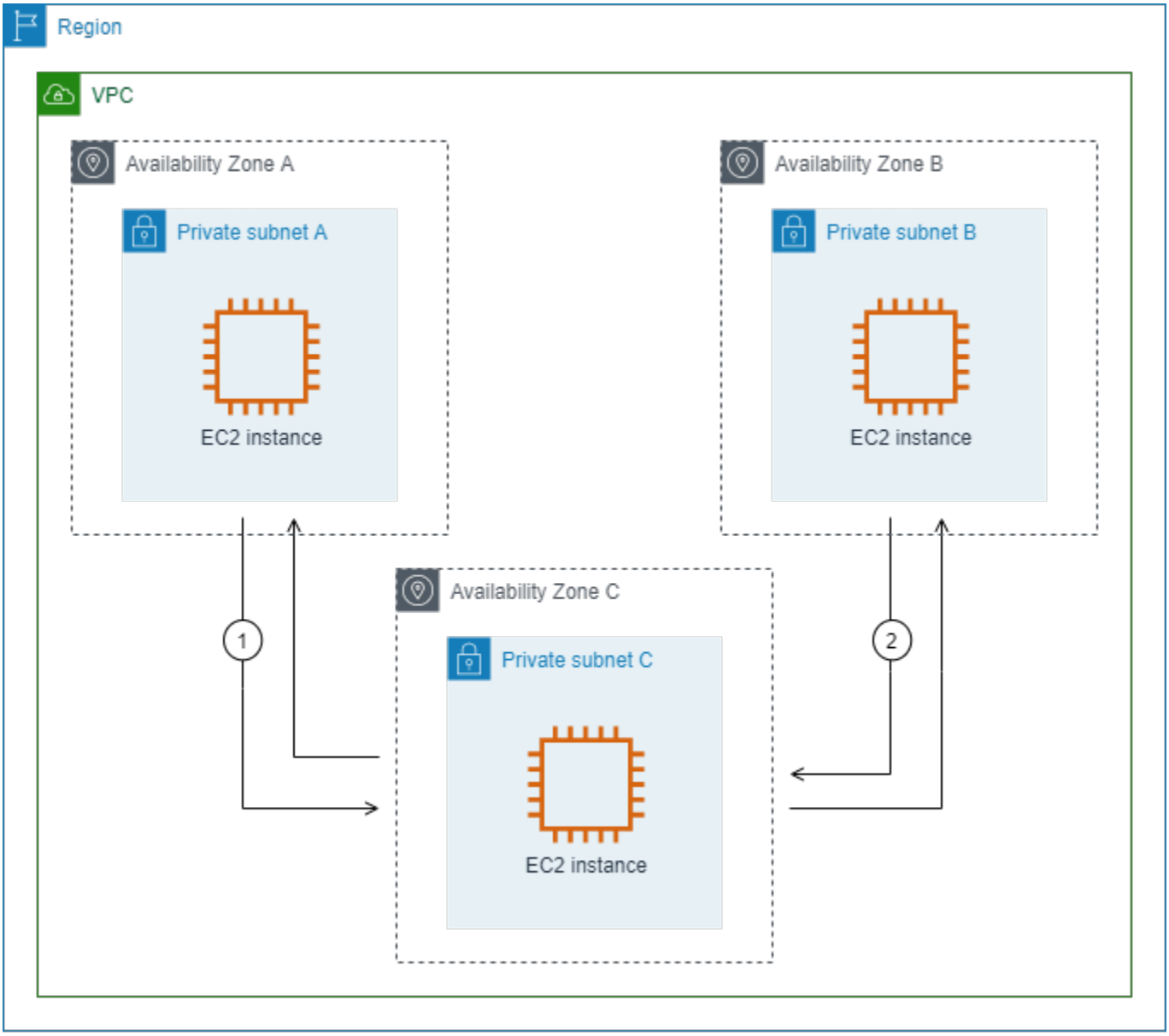
使用中间盒路由向导时，它将以下标签与路由表相关联：

- 键为“Origin”，值为“Middlebox wizard”
- 键为“date_created”，值为创建时间（例如“2021-02-18T22:25:49.137Z”）

检查子网之间的流量

请考虑如下场景，其中您在 VPC 中有多个子网，并且希望使用防火墙设备检查这些子网之间的流量。在 VPC 的单独子网中的 EC2 实例上配置并安装防火墙设备。

下图显示子网 C 中的 EC2 实例上安装的防火墙设备。此设备检查从子网 A 传输到子网 B（请参见 1）和从子网 B 传输到子网 A（请参见 2）的所有流量。



您使用 VPC 和中间盒子网的主路由表。子网 A 和 B 各有一个自定义路由表。

中间盒路由向导会自动执行以下操作：

- 创建路由表。
- 将必要的路由添加到新路由表中。
- 取消与子网关联的当前路由表的关联。
- 将中间盒路由向导创建的路由表与子网相关联。
- 创建一个标签，指示它是由中间盒路由向导创建的，并创建一个指示创建日期的标签。

中间盒路由向导不会修改现有的路由表。它会创建新的路由表，然后将它们与您的网关和子网资源相关联。如果您的资源已与现有路由表显式关联，则首先取消现有路由表的关联，然后将新路由表与您的资源相关联。您的现有路由表不会被删除。

如果不使用中间盒路由向导，则必须手动配置路由表，然后将路由表分配给子网和互联网网关。

自定义子网 A 的路由表

子网 A 的路由表具有以下路由。

目标位置	目标	用途
VPC CIDR	本地	本地路由
## B CIDR	*appliance-eni*	将发往子网 B 的流量路由到中间盒

使用中间盒路由向导时，它将以下标签与路由表相关联：

- 键为“Origin”，值为“Middlebox wizard”
- 键为“date_created”，值为创建时间（例如“2021-02-18T22:25:49.137Z”）

自定义子网 B 的路由表

子网 B 的路由表具有以下路由。

目标位置	目标	用途
VPC CIDR	本地	本地路由
## A CIDR	*appliance-eni*	将发往子网 A 的流量路由到中间盒

使用中间盒路由向导时，它将以下标签与路由表相关联：

- 键为“Origin”，值为“Middlebox wizard”
- 键为“date_created”，值为创建时间（例如“2021-02-18T22:25:49.137Z”）

主路由表

子网 C 使用主路由表。主路由表具有以下路由。

目标位置	目标	用途
VPC CIDR	本地	本地路由

使用中间盒路由向导时，它将以下标签与路由表相关联：

- 键为“Origin”，值为“Middlebox wizard”
- 键为“date_created”，值为创建时间（例如“2021-02-18T22:25:49.137Z”）

删除子网

如果不再需要某个子网，则可将其删除。如果该子网包含任何网络接口，则无法将其删除。例如，您必须首先终止子网中的所有实例，然后才能将其删除。

删除子网时，与该子网关联的 CIDR 块会返回到 VPC 的可用 IP 地址池中。这表示子网 CIDR 范围内的 IP 地址可以重新分配给同一 VPC 内的其他子网或资源。

务必注意，删除子网并不会自动删除其中的资源。您必须先终止任何 EC2 实例、删除所有网络接口并删除与子网关联的任何其他资源，才能继续删除子网。

使用控制台删除子网

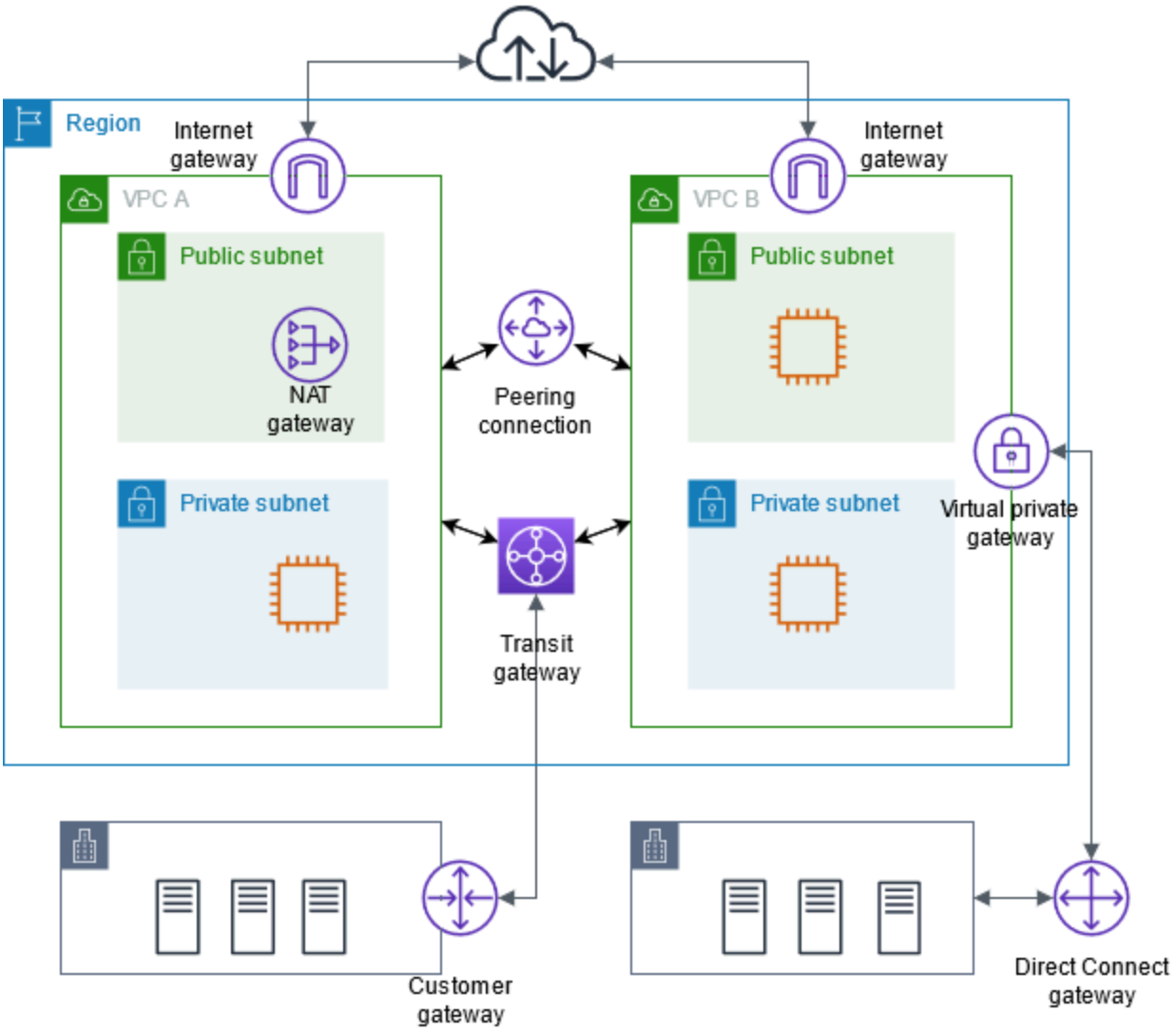
1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 终止子网中的所有实例。有关更多信息，请参阅《Amazon EC2 用户指南》中的 终止实例。
3. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
4. 在导航窗格中，选择 Subnets(子网)。
5. 选择子网，然后选择 Actions（操作）、Delete subnet（删除子网）。
6. 提示进行确认时，键入 **delete**，然后选择 Delete（删除）。

使用 AWS CLI 删除子网

使用 delete-subnet 命令。

将 VPC 连接到其他网络

您可以将虚拟私有云（VPC）连接到其他网络，例如其他 VPC、互联网或本地部署的网络。



您可以将虚拟私有云（VPC）连接到其他网络，例如其他 VPC、互联网或本地部署的网络。

该图演示了其中的一些连接选项。VPC A 通过互联网网关连接到互联网，而私有子网中的 EC2 实例可以使用公有子网中的 NAT 网关连接到互联网。VPC B 也连接到互联网，但通过直接互联网网关进行连接，这使得公有子网中的 EC2 实例能够访问互联网。

此外，VPC A 和 VPC B 通过 VPC 对等连接和中转网关相互连接。该中转网关具有连接至某个数据中心的 VPN 挂载，而 VPC B 则具有连接到同一数据中心的 AWS Direct Connect 连接。这种互连性让组织能够将自己的云资源与本地基础设施集成，从而创建混合云环境。

将 VPC 连接到其他网络是在 AWS 中构建云基础设施的重要方面。这为组织提供了灵活性和对其联网配置的控制权，让组织能够设计与自身业务要求和安全需求一致的 VPC 架构。这些连接选项有助于实现分布式 IT 环境的各个组件之间的高效数据流，无论组件是在云端还是在本地。

AWS 提供了许多工具和功能来实现这些 VPC 连接，包括互联网网关、NAT 网关、VPC 对等互连、中转网关和 AWS Direct Connect。通过利用这些功能，组织可以创建安全且集成的云环境，与其现有的 IT 基础设施无缝集成。

您可以将您的 Virtual Private Cloud（VPC）连接到其他网络。例如，其他 VPC、互联网或本地部署的网络。

有关更多信息，请参阅 Amazon Virtual Private Cloud 连接选项。

内容

- 使用互联网网关为 VPC 启用互联网访问
- 使用仅出口互联网网关允许出站 IPv6 流量
- 使用 NAT 设备连接到互联网或其他网络
- 将弹性 IP 地址关联到 VPC 中的资源
- 使用中转网关将您的 VPC 连接到其他 VPC 和网络
- 使用 AWS Virtual Private Network 将 VPC 连接到远程网络
- 使用 VPC 对等连接来连接 VPC

使用互联网网关为 VPC 启用互联网访问

互联网网关是一种横向扩展、冗余且高度可用的 VPC 组件，支持在 VPC 和 Internet 之间进行通信。它支持 IPv4 和 IPv6 流量。它不会对您的网络流量造成可用性风险或带宽限制。

借助互联网网关，公有子网中具有公有 IPv4 地址或 IPv6 地址的资源（例如 EC2 实例）可以连接到互联网。同样，互联网上的资源也可以使用公有 IPv4 地址或 IPv6 地址发起到子网中的资源的连接。例如，您可以通过互联网网关，使用本地电脑连接到 AWS 中的 EC2 实例。

互联网网关为您的 VPC 路由表中可通过互联网路由的流量提供目标。对于使用 IPv4 的通信，互联网网关还会执行网络地址转换 (NATI)。有关更多信息，请参阅 IP 地址和 NAT。

定价

互联网网关不收取任何费用，但您需要为使用互联网网关的 EC2 实例支付数据传输费用。有关更多信息，请参阅 Amazon EC2 按需定价。

内容

- 互联网网关基本信息
- 向子网添加互联网访问权限
- 删除互联网网关

互联网网关基本信息

要使用互联网网关，您必须将其连接到 VPC 并配置路由。

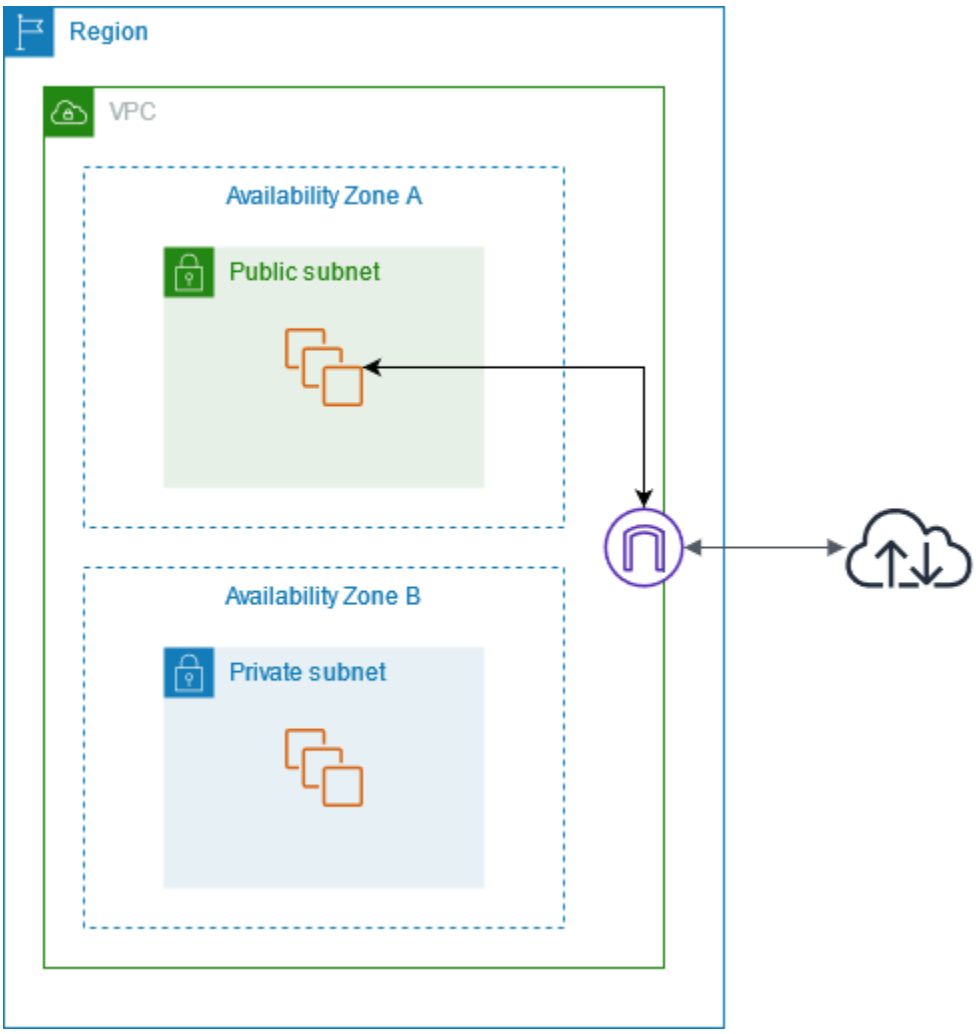
路由配置

如果子网的关联路由表包含指向互联网网关的路由，则该子网称为公有子网。如果子网的关联路由表没有指向互联网网关的路由，则该子网称为私有子网。

在公有子网路由表中，您可以将互联网网关的路由指定为路由表未明确知晓的所有目的地（对于 IPv4 为 0.0.0.0/0，对于 IPv6 为 ::/0）。或者，您也可以将路由范围设定为一个较小的 IP 地址范围，例如，公司在AWS以外的公有端点的公有 IPv4 地址，或 VPC 以外的其他 Amazon EC2 实例的弹性 IP 地址。

互联网网关图

在下图中，可用区 A 中的子网是公有子网，因为其路由表具有将所有互联网绑定的 IPv4 流量发送到互联网网关的路由。公有子网中的实例必须具有公有 IP 地址或弹性 IP 地址，才能通过互联网网关与互联网进行通信。为了进行比较，可用区 B 中的子网是私有子网，因为其路由表没有通往互联网网关的路由。因为没有到互联网网关的路由，所以私有子网中的实例即使具有公有 IP 地址，也无法与互联网通信。



IP 地址和 NAT

要为 IPv4 启用互联网通信，实例必须具有公有 IPv4 地址。您可以将 VPC 配置为自动向实例分配公有 IPv4 地址，也可以为实例分配弹性 IP 地址。实例只了解 VPC 和子网内定义的私有 (内部) IP 地址空间。互联网网关以逻辑方式代表实例提供一对一 NAT，这样一来，当流量离开 VPC 子网并流向 Internet 时，回复地址字段将设置为实例的公有 IPv4 地址或弹性 IP 地址，而不是私有 IP 地址。相反，指定发往实例的公有 IPv4 地址或弹性 IP 地址的流量会先将其目标地址转换为实例的私有 IPv4 地址，然后再传输到 VPC。

要为 IPv6 启用 Internet 通信，VPC 和子网必须具有关联的 IPv6 CIDR 块，并且必须为实例分配此子网范围内的 IPv6 地址。IPv6 地址是全球唯一的，因此默认为公有。

对默认和非默认 VPC 的 Internet 访问

下表概述了 VPC 是否自动提供通过 IPv4 或 IPv6 进行 Internet 访问所需的组件。

组件	默认 VPC	非默认 VPC
互联网网关	是	否
包含将 IPv4 流量路由到互联网网关的路由的路由表 (0.0.0.0/0)	是	否
包含将 IPv6 流量路由到互联网网关的路由的路由表 (::/0)	否	否
公有 IPv4 地址自动分配到在子网中启动的实例	是 (默认子网)	否 (非默认子网)
IPv6 地址自动分配到在子网中启动的实例	否 (默认子网)	否 (非默认子网)

向子网添加互联网访问权限

下面介绍如何使用互联网网关从非默认 VPC 中的子网访问互联网。您必须创建互联网网关，将其连接到 VPC，并为子网配置路由。

在为子网配置互联网访问后，您必须确保子网中的资源可以访问互联网。例如，EC2 实例必须具有公有 IPv4 或 IPv6 地址，并且实例的安全组必须允许进出互联网的特定流量。

或者，要为实例提供互联网访问，而不为其分配公有 IP 地址，请改用 NAT 设备。有关更多信息，请参阅 NAT 设备。

要删除互联网访问权限，您可以将互联网网关与 VPC 分离，然后将其删除。有关更多信息，请参阅 the section called “删除互联网网关”。

任务

- 步骤 1：创建互联网网关
- 步骤 2：将互联网网关连接到 VPC
- 步骤 3：向子网路由表添加路由

步骤 1：创建互联网网关

请按照以下步骤创建互联网网关。

使用控制台创建互联网网关

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Internet gateways（互联网网关）。
3. 选择创建互联网网关。
4. （可选）输入互联网网关的名称。
5. （可选）若要添加标签，请选择 Add new tag（添加新标签），然后输入该标签的键和值。
6. 选择创建互联网网关。
7. （可选）要立即将互联网网关附加到 VPC，请从屏幕顶部的横幅中选择附加到 VPC，选择可用的 VPC，然后选择连接互联网网关。您也可以在其他时间将互联网网关附加到 VPC。

使用命令行创建互联网网关

- create-internet-gateway (AWS CLI)
- New-EC2InternetGateway (AWS Tools for Windows PowerShell)

步骤 2：将互联网网关连接到 VPC

使用互联网网关之前，必须将其附加到 VPC。

使用控制台将互联网网关附加到 VPC

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Internet gateways（互联网网关）。
3. 选中互联网网关的复选框。
4. 要附加互联网网关，依次选择操作、“附加到 VPC”，选择可用的 VPC，然后选择附加互联网网关。
5. 要将互联网网关分离，依次选择操作、从 VPC 分离，然后选择分离互联网网关。当系统提示进行确认时，选择分离互联网网关。

使用命令行将互联网网关附加到 VPC

- attach-internet-gateway (AWS CLI)
- Add-EC2InternetGateway (AWS Tools for Windows PowerShell)

步骤 3：向子网路由表添加路由

子网的路由表必须有一个将互联网流量发送到互联网网关的路由。

使用控制台配置子网路由表

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Route tables（路由表）。
3. 选择子网的路由表。默认情况下，子网使用 VPC 的主路由表。或者，您可以创建路由表，然后将子网与新的路由表关联。
4. 在路由选项卡中选择编辑路由，然后选择添加路由。
5. 为目的地输入 0.0.0.0/0，并为目标选择互联网网关。
6. 选择保存更改。

使用命令行配置子网路由表

- create-route (AWS CLI)
- New-EC2Route (AWS Tools for Windows PowerShell)

删除互联网网关

如果不再需要 VPC 的互联网访问，您可以将互联网网关与 VPC 分离，然后将其删除。无法删除仍附加到 VPC 的互联网网关。如果 VPC 的某些资源具有关联的公有 IP 地址或弹性 IP 地址，则无法分离互联网网关。

使用控制台将互联网网关与 VPC 分离

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Internet gateways（互联网网关）。
3. 选中互联网网关的复选框。

4. 要附加互联网网关，依次选择操作、“附加到 VPC”，选择可用的 VPC，然后选择附加互联网网关。
5. 要将互联网网关分离，依次选择操作、从 VPC 分离，然后选择分离互联网网关。当系统提示进行确认时，选择分离互联网网关。

使用命令行描述互联网网关（包括连接）

- describe-internet-gateways (AWS CLI)
- Get-EC2InternetGateway (AWS Tools for Windows PowerShell)

使用命令行将互联网网关与 VPC 分离

- detach-internet-gateway (AWS CLI)
- Dismount-EC2InternetGateway (AWS Tools for Windows PowerShell)

使用控制台删除互联网网关

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Internet gateways（互联网网关）。
3. 选中互联网网关的复选框。
4. 依次选择操作、删除互联网网关。
5. 当系统提示进行确认时，输入 **delete**，然后选择删除互联网网关。

使用命令行删除互联网网关

- delete-internet-gateway (AWS CLI)
- Remove-EC2InternetGateway (AWS Tools for Windows PowerShell)

使用仅出口互联网网关允许出站 IPv6 流量

仅出口互联网网关是一种横向扩展、支持冗余且高度可用的 VPC 组件，它能够实现从 VPC 中的实例经由 IPv6 到 Internet 的出站通信，并防止 Internet 发起与您的实例的 IPv6 连接。

仅出口互联网网关只适用于 IPv6 流量。要通过 IPv4 实现仅出站 Internet 通信，请改用 NAT 网关。有关更多信息，请参阅 NAT 网关。

定价

仅出口互联网网关不收取任何费用，但您需要为使用互联网网关的 EC2 实例支付数据传输费用。有关更多信息，请参阅 Amazon EC2 按需定价。

内容

- 仅出口互联网网关基础知识
- 向子网添加仅出口互联网访问

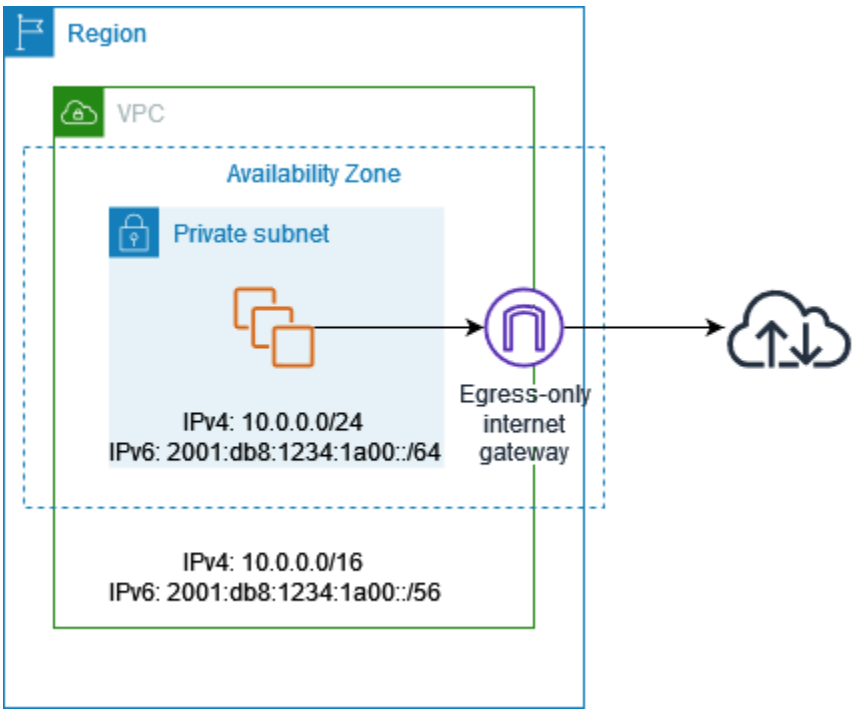
仅出口互联网网关基础知识

IPv6 地址是全球唯一的，因此默认为公有。如果您希望实例能够访问 Internet，但又想要阻止 Internet 上的资源发起与您的实例的通信，则您可以使用仅出口互联网网关。为此，请在 VPC 中创建一个仅出口互联网网关，然后向路由表中添加一条将所有 IPv6 流量 (`::/0`) 或特定的 IPv6 地址范围指向仅出口互联网网关的路由。子网中与路由表关联的 IPv6 流量会被路由到仅出口互联网网关。

仅出口互联网网关是有状态的：它将来自子网中实例的流量转发到 Internet 或其他 AWS 服务，然后将响应发回给实例。

您无法将安全组与仅出口互联网网关关联，以控制允许访问或离开仅出口互联网网关的流量。您可以使用网络 ACL 控制仅出口互联网网关路由的进出子网的流量。

在下图中，VPC 同时具有 IPv4 和 IPv6 CIDR 块，子网同时具有 IPv4 和 IPv6 CIDR 块。此 VPC 拥有一个仅出口互联网网关。



以下是与该子网关联的路由表示例。有一条路由会将所有传出至互联网的所有 IPv6 流量（::/0）发送到仅出口互联网网关。

目标位置	目标
10.0.0.0/16	本地
2001:db8:1234:1a00:/64	本地
::/0	*eigw-id*

向子网添加仅出口互联网访问

以下任务介绍如何为私有子网创建仅出口（出站）互联网网关，以及如何为该子网配置路由。

任务

- 1. 创建仅出口互联网网关
- 2. 创建自定义路由表
- 3. 删除仅出口互联网网关
- 命令行概述

1. 创建仅出口互联网网关

您可以使用 Amazon VPC 控制台为您的 VPC 创建一个仅出口互联网网关。

创建仅出口互联网网关

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Egress Only Internet Gateways。
3. 选择创建仅出口互联网网关。
4. （可选）添加或删除标签。

   [添加标签] 选择添加新标签，然后执行以下操作：

   - 对于 Key（键），输入键名称。
   - 对于值，输入键值。

   [删除标签] 选择标签的“键”和“值”右侧的删除。
5. 选择要在其中创建仅出口互联网网关的 VPC。
6. 选择创建。

2. 创建自定义路由表

要将发往 VPC 外部的流量发送到仅出口互联网网关，您必须创建一个自定义路由表并添加将流量发送到该网关的路由，然后将其与您的子网关联。

创建自定义路由表并添加到仅出口互联网网关的路由

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，依次选择 Route Tables (路由表)、Create Route Table (创建路由表)。
3. 在 Create Route Table (创建路由表) 对话框中，可以选择命名您的路由表，选择您的 VPC，然后选择 Yes, Create (是，创建)。
4. 选择您刚刚创建的自定义路由表。详细信息窗格中会显示选项卡，以供您使用其路径、关联和路线传播。
5. 在 Routes (路由) 选项卡中，选择 Edit routes (编辑路由)，在 Destination (目的地) 框中指定 `::/0`，从 Target (目标) 列表中选择仅出口互联网网关 ID，然后选择 Save changes (保存更改)。

6. 在 Subnet associations (子网关联) 选项卡上，选择 Edit subnet associations (编辑子网关联)，然后选中子网对应的复选框。选择 Save。

或者，您也可以向与您的子网关联的现有路由表添加路由。选择您现有的路由表，然后按照上述步骤 5 和 6 为仅出口互联网网关添加路由。

有关路由表的更多信息，请参见配置路由表。

3. 删除仅出口互联网网关

如果您不再需要某一仅出口互联网网关，则可将其删除。路由表中指向已删除的仅出口互联网网关的任何路由都将保持 `blackhole` 状态，直到您手动删除或更新路由。

删除仅出口互联网网关

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择仅出口互联网网关，然后选择仅出口互联网网关。
3. 选择 Delete。
4. 在确认对话框中选择删除仅出口互联网网关。

命令行概述

您可以使用命令行执行此页面上介绍的任务。

创建仅出口互联网网关

- create-egress-only-internet-gateway (AWS CLI)
- New-EC2EgressOnlyInternetGateway (AWS Tools for Windows PowerShell)

描述仅出口互联网网关

- describe-egress-only-internet-gateways (AWS CLI)
- Get-EC2EgressOnlyInternetGatewayList (AWS Tools for Windows PowerShell)

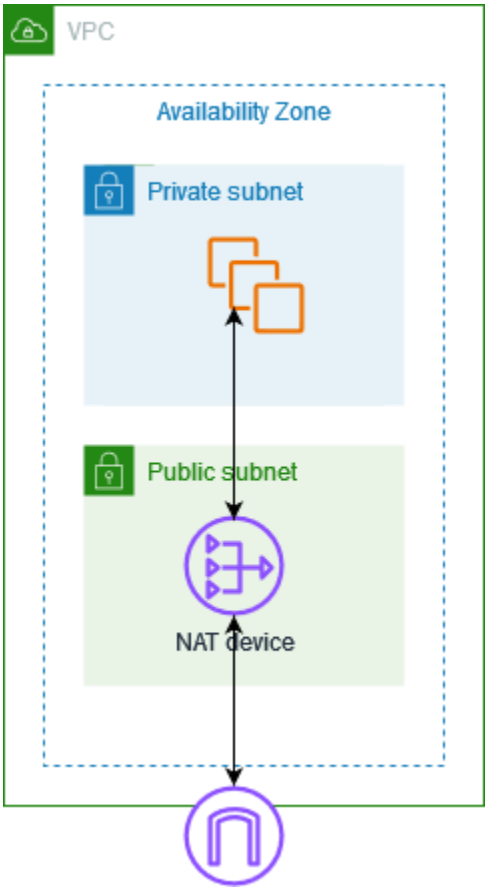
删除仅出口互联网网关

- delete-egress-only-internet-gateway (AWS CLI)
- Remove-EC2EgressOnlyInternetGateway (AWS Tools for Windows PowerShell)

使用 NAT 设备连接到互联网或其他网络

您可以使用 NAT 设备允许私有子网中的资源连接到互联网、其他 VPC 或本地网络。这些实例可以与 VPC 外部的服务进行通信，但它们无法接收未经请求的连接请求。

例如，下图显示了公有子网中的 NAT 设备，该设备允许私有子网中的 EC2 实例通过互联网网关连接到互联网。NAT 设备将实例的源 IPv4 地址替换为 NAT 设备的地址。向实例发送响应流量时，NAT 设备会将地址转换回原始源 IPv4 地址。



Important

- 我们在本文中使用 NAT 是为了遵循通行的 IT 做法，而 NAT 设备的实际作用包括地址转换和端口地址转换 (PAT) 两方面。
- 您可以使用 AWS 提供的托管式 NAT 设备（称为 NAT 网关），也可以在 EC2 实例（称为 NAT 实例）中创建自己的 NAT 设备。我们建议您使用 NAT 网关，因为它们提供了更好的可用性和带宽，而且管理工作所需的工作量更少。

内容

- NAT 网关
- NAT 实例
- 比较 NAT 网关和 NAT 实例

NAT 网关

NAT 网关是一种网络地址转换 (NATI) 服务。您可以使用 NAT 网关，以便私有子网中的实例可以连接到 VPC 外部的服务，但外部服务无法启动与这些实例的连接。

在创建 NAT 网关时，您指定以下连接类型之一：

- 公开：（默认）私有子网中的实例可以通过公共 NAT 网关连接到互联网，但这些实例不能接收来自互联网的、未经请求的入站连接。您在公有子网中创建公有 NAT 网关，并且必须在创建时将弹性 IP 地址与 NAT 网关相关联。您可以将流量从 NAT 网关路由到 VPC 的互联网网关。或者，您可以使用公有 NAT 网关连接到其他 VPC 或本地部署网络。在这种情况下，您可以借助中转网关或虚拟私有网关路由来自 NAT 网关的流量。
- 私有：私有子网中的实例可以通过私有 NAT 网关连接到其他 VPC 或您的本地网络，但这些实例无法接收来自其他 VPC 或本地网络的、未经请求的入站连接。您可以借助中转网关或虚拟私有网关路由来自 NAT 网关的流量。您不能将弹性 IP 地址与私有 NAT 网关相关联。您可以将互联网网关连接到具有私有 NAT 网关的 VPC，但如果您将流量从私有 NAT 网关路由到互联网网关，则互联网网关会丢弃流量。

NAT 网关适用于 IPv4 或 IPv6 流量（使用 DNS64 和 NAT64 ）。通过 IPv6 实现仅出站互联网通信的另一种选择是使用仅出口互联网网关。

私有和公有 NAT 网关都会将实例的源私有 IPv4 地址映射到 NAT 网关的私有 IPv4 地址，但是对于公有 NAT 网关，互联网网关随后会将公有 NAT 网关的私有 IPv4 地址映射到与 NAT 网关关联的弹性 IP 地址。将响应流量发送到实例时，无论是使用公有还是私有 NAT 网关，NAT 网关都会将地址转换回原始源 IP 地址。

> **Important**
>
> 必须始终从包含 NAT 网关的 VPC 内启动连接。
> 您可以使用公有或私有 NAT 网关将流量路由到传输网关和虚拟私有网关。
> 如果您使用私有 NAT 网关连接到传输网关或虚拟私有网关，则到达目的地的流量将来自私有 NAT 网关的私有 IP 地址。

如果您使用公有 NAT 网关连接到传输网关或虚拟私有网关，则到达目的地的流量将来自公有 NAT 网关的私有 IP 地址。仅当与同一 VPC 中的互联网网关结合使用时，公有 NAT 网关才会使用其 EIP 作为源 IP 地址。
NAT 网关可支持最大传输单位（MTU）为 8500 的流量。有关更多信息，请参阅 NAT 网关基础知识。

内容

- NAT 网关基础知识
- 使用 NAT 网关
- NAT 网关使用案例
- DNS64 和 NAT64
- 使用 Amazon CloudWatch 监控 NAT 网关
- 排查 NAT 网关的问题
- 适用于 NAT 网关的定价

NAT 网关基础知识

每个 NAT 网关都在特定可用区中创建，并在该可用区进行冗余实施。您可以在每个可用区中创建的 NAT 网关存在数量配额。有关更多信息，请参阅 Amazon VPC 配额。

如果您在多个可用区中拥有资源并且它们共享一个 NAT 网关，如果该 NAT 网关的可用区不可用，其他可用区中的资源将无法访问 Internet。为提高故障恢复能力，请在每个可用区中创建一个 NAT 网关，并配置路由以确保这些资源使用自身可用区中的 NAT 网关。

以下特征和规则适用于 NAT 网关：

- NAT 网关支持以下协议：TCP、UDP 和 ICMP。
- IPv4 或 IPv6 流量支持 NAT 网关。对于 IPv6 流量，NAT 网关将执行 NAT64。通过与 DNS64 结合使用（在 Route 53 Resolver 上可用），Amazon VPC 子网中的 IPv6 工作负载可以与 IPv4 资源进行通信。这些 IPv4 服务可能存在于同一 VPC（在单独子网中）或其他 VPC、本地环境或互联网上。
- NAT 网关支持 5 Gbps 带宽并会自动扩展到 100 Gbps。如果您需要更大的带宽，您可以将资源拆分到多个子网中，并在每个子网中创建 NAT 网关。

- 一个 NAT 网关每秒能处理 1 百万个数据包，还能自动扩展到每秒 1 千万个数据包。超出此限制后，NAT 网关将丢弃数据包。为防止数据包丢失，请将资源拆分到多个子网中，并为每个子网中创建单独的 NAT 网关。
- 对于每个唯一目标，每个 IPv4 地址最多可以支持 55000 个并发连接。唯一目标由目标 IP 地址、目标端口和协议（TCP/UDP/ICMP）的唯一组合标识。您可以通过将最多 8 个 IPv4 地址（1 个主要 IPv4 地址和 7 个辅助 IPv4 地址）关联到 NAT 网关来提高此限制。默认情况下，公有 NAT 网关只能关联 2 个弹性 IP 地址。您可以通过请求调整限额来提高此限制。有关更多信息，请参阅 弹性 IP 地址。
- 您可以选择分配给 NAT 网关的私有 IPv4 地址，也可以从子网的 IPv4 地址范围中自动分配。在删除私有 NAT 网关之前，所分配的私有 IPv4 地址将一直存在。您无法分离私有 IPv4 地址，也无法连接其他私有 IPv4 地址。
- 您不能将安全组与 NAT 网关关联。您可以将安全组与实例相关联，以控制入站和出站流量。
- 您可以使用网络 ACL 控制进出 NAT 网关所在子网的流量。NAT 网关使用端口 1024–65535。有关更多信息，请参阅 使用网络访问控制列表控制子网流量。
- NAT 网关会收到一个网络接口。您可以选择分配给接口的私有 IPv4 地址，也可以从子网的 IPv4 地址范围中自动分配。您可以在 Amazon EC2 控制台中查看 NAT 网关的网络接口。有关更多信息，请参阅查看有关网络接口的详细信息。您不能修改此网络接口的属性。
- 无法通过 VPC 对等连接将流量路由到 NAT 网关。
- 您无法使用虚拟专用网关将流量从 Site-to-Site VPN 或 Direct Connect 路由到 NAT 网关。如果您使用中转网关而不是虚拟专用网关，则可以将流量从 Site-to-Site VPN 或 Direct Connect 路由到 NAT 网关。
- NAT 网关支持最大传输单位（MTU）为 8500 的流量，但请务必注意以下几点：
  - 网络连接的 MTU 是能够通过该连接传递的最大可允许数据包的大小（以字节为单位）。连接的 MTU 越大，可在单个数据包中传递的数据越多。
  - 将丢弃到达 NAT 网关的大小超过 8500 字节的数据包（如果适用，则将被分段）。
  - 为防止在使用公有 NAT 网关通过互联网与资源通信时可能发生的数据包丢失，EC2 实例的 MTU 设置不应超过 1500 字节。有关检查和设置实例 MTU 的更多信息，请参阅《Amazon EC2 用户指南》中的在您的 Linux 实例上检查并设置 MTU。
  - NAT 网关通过 FRAG_NEEDED ICMPv4 数据包和 Packet Too Big（PTB）ICMPv6 数据包支持路径 MTU 发现（PMTUD）。
  - NAT 网关会对所有数据包强制执行最大分段大小（MSS）固定。有关更多信息，请参阅 RFC879。

使用 NAT 网关

您可以使用 Amazon VPC 控制台创建和管理 NAT 网关。

任务

- 控制 NAT 网关的使用
- 创建 NAT 网关
- 编辑辅助 IP 地址关联
- 标记 NAT 网关
- 删除 NAT 网关
- 命令行概述

控制 NAT 网关的使用

默认情况下，用户无权使用 NAT 网关。您可以创建一个 IAM 角色，并向该角色附加一个向用户授予 NAT 网关创建、描述和删除权限的策略。有关更多信息，请参阅 适用于 Amazon VPC 的 Identity and Access Management。

创建 NAT 网关

请按照以下过程创建 NAT 网关。

相关限额

- 如果您已耗尽分配给账户的 EIP 数量，则将无法创建公有 NAT 网关。有关 EIP 配额以及如何调整配额的更多信息，请参阅 弹性 IP 地址。
- 私有 NAT 网关最多可以分配 8 个私有 IPv4 地址。此限制不可调整。
- 默认情况下，公有 NAT 网关只能关联 2 个弹性 IP 地址。您可以通过请求调整限额来提高此限制。有关更多信息，请参阅 弹性 IP 地址。

创建 NAT 网关

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 NAT 网关。
3. 选择创建 NAT 网关。

4. （可选）指定 NAT 网关的名称。这将创建一个标签，其中键为 **Name**，值是您指定的名称。
5. 选择要在其中创建 NAT 网关的子网。
6. 对于连接类型，保持默认的公有选择不变，以创建公有 NAT 网关，或者选择私有，以创建私有 NAT 网关。有关公有和私有 NAT 网关之间差异的更多信息，请参阅 NAT 网关。
7. 如果您选择公有，请执行以下操作；否则，请跳至第 8 步：

    1. 选择弹性 IP 分配 ID，为 NAT 网关分配 EIP，或者选择分配弹性 IP，为公有 NAT 网关自动分配弹性 EIP。默认情况下，公有 NAT 网关只能关联 2 个弹性 IP 地址。您可以通过请求调整限额来提高此限制。有关更多信息，请参阅 弹性 IP 地址。

    > **Important**
    >
    > 当您为公有 NAT 网关分配 EIP 时，EIP 的网络边界组必须与您启动公有 NAT 网关的可用区（AZ）的网络边界组相匹配。如果不匹配，NAT 网关将无法启动。可以通过查看子网的详细信息来了解子网可用区的网络边界组。同样，可以通过查看 EIP 地址的详细信息来了解 EIP 的网络边界组。有关网络边界组和 EIP 的更多信息，请参阅 1. 分配弹性 IP 地址。

    2. （可选）选择其他设置，然后在私有 IP 地址 – 可选下，为 NAT 网关输入私有 IPv4 地址。如果不输入地址，AWS 会自动从 NAT 网关所在的子网中为 NAT 网关随机分配一个私有 IPv4 地址。
    3. 跳至步骤 11。
8. 如果您选择私有，请选择其他设置，然后在私有 IP 地址分配方法下，选择下列选项中的一种：

    - 自动分配：AWS 将为 NAT 网关选择主私有 IPv4 地址。对于自动分配的私有 IPv4 地址数量，您可以选择为 NAT 网关指定辅助私有 IPv4 地址的数量。AWS 将从 NAT 网关的子网中随机选择这些 IP 地址。
    - 自定义：对于主私有 IPv4 地址，选择 NAT 网关的主私有 IPv4 地址。对于辅助私有 IPv4 地址，您可以选择为 NAT 网关指定最多 7 个辅助私有 IPv4 地址。
9. 如果您在步骤 8 中选择了自定义，请跳过此步骤。如果您选择了自动分配，请在自动分配的私有 IP 地址数量下，选择您希望 AWS 分配给该私有 NAT 网关的辅助 IPv4 地址数量。最多可以选择 7 个 IPv4 地址。

    > **Note**
    >
    > 辅助 IPv4 地址是可选的，如果使用 NAT 网关的工作负载与单个目标的并发连接超过 55,000 个（相同的目标 IP、目标端口和协议），则应当分配辅助 IPv4 地址。辅助 IPv4

地址增加了可用的端口数量，从而提高了工作负载使用 NAT 网关建立连接的并发连接数限制。

10. 如果您在步骤 9 中选择了自动分配，请跳过此步骤。如果您选择了自定义，请执行以下操作：
    1. 在主要私有 IPv4 地址下，输入私有 IPv4 地址。
    2. 在辅助私有 IPv4 地址下，输入辅助私有 IPv4 地址，不超过 7 个。
11. （可选）若要向 NAT 网关添加标签，请选择 Add new tag（添加新标签），然后输入该标签的键和值。最多可以添加 50 个标签。
12. 选择创建 NAT 网关。
13. NAT 网关的初始状态为 `Pending`。状态更改为 `Available` 后，NAT 网关即可供您使用。请务必按需新路由表。有关示例，请参阅 the section called “使用案例”。

如果 NAT 网关的状态变成 `Failed`，则表示在创建过程中发生了错误。有关更多信息，请参阅 NAT 网关创建失败。

编辑辅助 IP 地址关联

对于每个唯一目标，每个 IPv4 地址最多可以支持 55000 个并发连接。唯一目标由目标 IP 地址、目标端口和协议（TCP/UDP/ICMP）的唯一组合标识。您可以通过将最多 8 个 IPv4 地址（1 个主要 IPv4 地址和 7 个辅助 IPv4 地址）关联到 NAT 网关来提高此限制。默认情况下，公有 NAT 网关只能关联 2 个弹性 IP 地址。您可以通过请求调整限额来提高此限制。有关更多信息，请参阅 弹性 IP 地址。

您可以使用 NAT 网关 CloudWatch 指标 ErrorPortAllocation 和 PacketsDropCount 来确定 NAT 网关是否正在生成端口分配错误或丢弃数据包。要解决此问题，请将辅助 IPv4 地址添加到 NAT 网关。

注意事项

- 您可以在创建私有 NAT 网关时或在使用本部分中的步骤创建 NAT 网关后添加辅助私有 IPv4 地址。只有在使用本部分中的步骤创建 NAT 网关后，才能将辅助 EIP 地址添加到公有 NAT 网关。
- NAT 网关最多可以关联 8 个 IPv4 地址（1 个主要 IPv4 地址和 7 个辅助 IPv4 地址）。私有 NAT 网关最多可以分配 8 个私有 IPv4 地址。默认情况下，公有 NAT 网关只能关联 2 个弹性 IP 地址。您可以通过请求调整限额来提高此限制。有关更多信息，请参阅 弹性 IP 地址。

编辑辅助 IPv4 地址关联

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。

2. 在导航窗格中，选择 NAT 网关。
3. 选择要编辑其辅助 IPv4 地址关联的 NAT 网关。
4. 选择操作，然后选择编辑辅助 IP 地址关联。
5. 如果要编辑私有 NAT 网关的辅助 IPv4 地址关联，请在操作下，选择分配新 IPv4 地址或取消分配现有 IPv4 地址。如果要编辑公有 NAT 网关的辅助 IPv4 地址关联，请在操作下，选择关联新 IPv4 地址或取消关联现有 IPv4 地址。
6. 请执行以下操作之一：

   - 如果您选择分配或关联新 IPv4 地址，请执行以下操作：

     1. 这个步骤为必填项。您必须选择一个私有 IPv4 地址。选择私有 IPv4 地址分配方法：

        - 自动分配：AWS 会自动选择主要私有 IPv4 地址。如果您希望 AWS 将最多 7 个辅助私有 IPv4 地址分配给 NAT 网关，则可以选择该选项。AWS 会自动从 NAT 网关所在的子网中随机选择并进行分配。
        - 自定义：选择要分配给 NAT 网关的主要私有 IPv4 地址和最多 7 个辅助私有 IPv4 地址。

     2. 在弹性 IP 分配 ID 下，选择要添加为辅助 IPv4 地址的 EIP。这个步骤为必填项。您必须选择一个 EIP 以及一个私有 IPv4 地址。如果您为私有 IP 地址分配方法选择了自定义，则还必须为您添加的每个 EIP 输入一个私有 IPv4 地址。

     > **Important**
     >
     > 当您为公有 NAT 网关分配辅助 EIP 时，该 EIP 的网络边界组必须与公有 NAT 网关所在可用区（AZ）的网络边界组匹配。如果不匹配，EIP 将无法分配。可以通过查看子网的详细信息来了解子网可用区的网络边界组。同样，可以通过查看 EIP 地址的详细信息来了解 EIP 的网络边界组。有关网络边界组和 EIP 的更多信息，请参阅 1. 分配弹性 IP 地址。

     NAT 网关最多可以关联 8 个 IP 地址。如果这是公有 NAT 网关，则每个区域的 EIP 都有默认限额限制。有关更多信息，请参阅 弹性 IP 地址。

   - 如果您选择取消分配或取消关联新 IPv4 地址，请完成以下操作：

     1. 在要取消分配的现有辅助 IP 地址下，选择要取消分配的辅助 IP 地址。
     2. （可选）在连接耗尽持续时间下，输入连接仍在进行时强制释放 IP 地址之前的最长等待时间（以秒为单位）。如果不输入值，则默认值为 350 秒。

7. 选择保存更改。

如果 NAT 网关的状态变成 `Failed`，则表示在创建过程中发生了错误。有关更多信息，请参阅 NAT 网关创建失败。

标记 NAT 网关

您可以对 NAT 网关进行标记，以帮助您识别它或根据组织的需要对其进行分类。有关使用标签的信息，请参阅《Amazon EC2 用户指南》中的标记您的 Amazon EC2 资源。

对于 NAT 网关支持成本分配标签。因此，您还可以使用标签来整理 AWS 账单并反映您自己的成本结构。有关更多信息，请参阅 AWS Billing 用户指南中的使用成本分配标签。有关设置包含标签的成本分配报告的更多信息，请参阅关于 AWS 账户账单中的月度成本分配报告。

标记 NAT 网关

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 NAT 网关。
3. 选择要标记的 NAT 网关，然后选择操作。然后选择管理标签。
4. 选择添加新标签，并定义标签的键和值。最多可以添加 50 个标签。
5. 选择保存。

删除 NAT 网关

您可以删除不再需要的 NAT 网关。删除 NAT 网关之后，其条目在一小时左右在 Amazon VPC 控制台中保持可见，在此之后自动删除。您无法自己删除此条目。

删除 NAT 网关会解除其弹性 IP 地址关联，但不会从您的账户释放该地址。如果删除 NAT 网关，则 NAT 网关路由会保留为 `blackhole` 状态，直到您删除或更新这些路由。

删除 NAT 网关

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 NAT 网关。
3. 选择 NAT 网关对应的单选按钮，然后选择 Actions（操作）、Delete NAT gateway（删除 NAT 网关）。
4. 提示进行确认时，输入 **`delete`**，然后选择 Delete（删除）。
5. 如果您不再需要与公有 NAT 网关关联的弹性 IP 地址，建议您释放该地址。有关更多信息，请参阅 5. 释放弹性 IP 地址。

命令行概述

您可以使用命令行执行此页面上介绍的任务。

将私有 IPv4 地址分配给私有 NAT 网关

- assign-private-nat-gateway-address（AWS CLI）
- Register-EC2PrivateNatGatewayAddress（AWS Tools for Windows PowerShell）

将弹性 IP 地址（EIP）和私有 IPv4 地址与公有 NAT 网关相关联

- associate-nat-gateway-address（AWS CLI）
- Register-EC2NatGatewayAddress（AWS Tools for Windows PowerShell）

创建 NAT 网关

- create-nat-gateway (AWS CLI)
- New-EC2NatGateway (AWS Tools for Windows PowerShell)

删除 NAT 网关

- delete-nat-gateway (AWS CLI)
- Remove-EC2NatGateway (AWS Tools for Windows PowerShell)

描述 NAT 网关

- describe-nat-gateways (AWS CLI)
- Get-EC2NatGateway (AWS Tools for Windows PowerShell)

将辅助弹性 IP 地址（EIP）与公有 NAT 网关取消关联

- disassociate-nat-gateway-address（AWS CLI）
- Unregister-EC2NatGatewayAddress（AWS Tools for Windows PowerShell）

标记 NAT 网关

- create-tags（AWS CLI）

- New-EC2Tag（AWS Tools for Windows PowerShell）

从私有 NAT 网关取消分配辅助 IPv4 地址

- unassign-private-nat-gateway-address（AWS CLI）
- Unregister-EC2PrivateNatGatewayAddress（AWS Tools for Windows PowerShell）

NAT 网关使用案例

以下是公有和私有 NAT 网关的使用案例示例。

场景

- 从私有子网访问互联网
- 从允许列出的 IP 地址访问您的网络
- 实现重叠网络之间的通信

从私有子网访问互联网

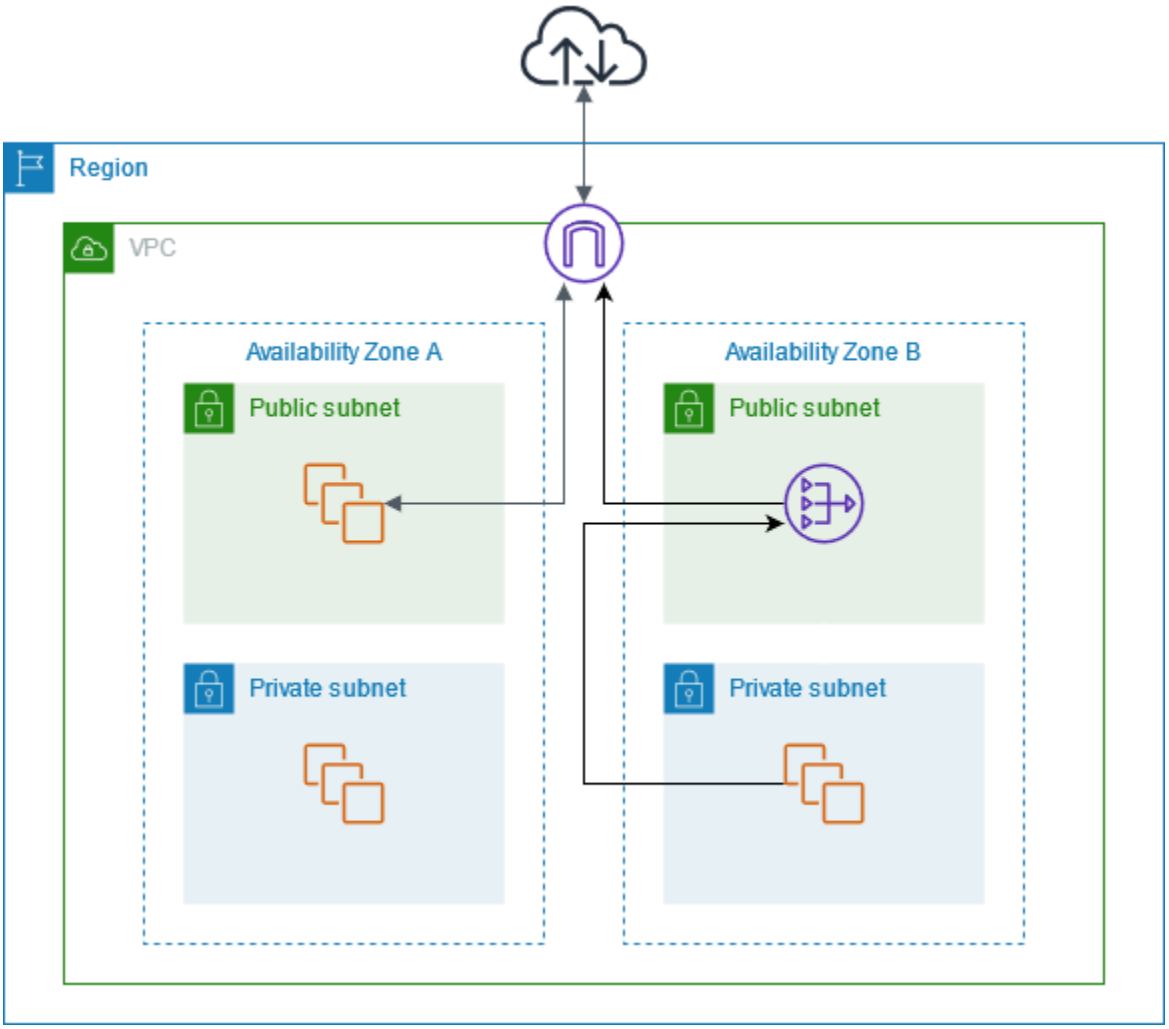
您可以使用公有 NAT 网关，以启用私有子网中的实例，将出站流量发送到互联网，同时防止互联网与这些实例建立连接。

内容

- 概览
- 路由
- 测试公有 NAT 网关

概览

下图阐明了此使用案例。有两个可用区，每个可用区都有两个子网。每个子网的路由表都决定了流量的路由方式。在可用区 A 中，公有子网中的实例可以通过到互联网网关的路由访问互联网，而私有子网中的实例没有到互联网的路由。在可用区 B 中，公有子网包含一个 NAT 网关，私有子网中的实例可以通过到公有子网中的 NAT 网关的路由来访问互联网。私有和公有 NAT 网关都会将实例的源私有 IPv4 地址映射到私有 NAT 网关的私有 IPv4 地址，但是对于公有 NAT 网关，互联网网关随后会将公有 NAT 网关的私有 IPv4 地址映射到与 NAT 网关关联的弹性 IP 地址。将响应流量发送到实例时，无论是使用公有还是私有 NAT 网关，NAT 网关都会将地址转换回原始源 IP 地址。



请注意，如果可用区 A 中私有子网内的实例也需要访问互联网，您可以创建从该子网到可用区 B 中 NAT 网关的路由。或者，您也可以在包含需要访问互联网的资源的每个可用区创建一个 NAT 网关来提高故障恢复能力。有关示例图，请参见 the section called “私有服务器”。

路由

以下是可用区 A 中与公有子网关联的路由表。第一个条目是本地路由；它使子网中的实例能够使用私有 IP 地址与 VPC 中的其他实例进行通信。第二个条目将所有其他子网流量发送到互联网网关，从而使子网中的实例能够访问互联网。

目标位置	目标
VPC CIDR	本地
0.0.0.0/0	*internet-gateway-id*

以下是可用区 A 中与私有子网关联的路由表。该条目是本地路由，它使子网中的实例能够使用私有 IP 地址与 VPC 中的其他实例进行通信。此子网中的实例无法访问互联网。

目标位置	目标
VPC CIDR	本地

以下是可用区 B 中与公有子网关联的路由表。第一个条目是本地路由，它使子网中的实例能够使用私有 IP 地址与 VPC 中的其他实例进行通信。第二个条目将所有其他子网流量发送到互联网网关，从而使子网中的 NAT 网关能够访问互联网。

目标位置	目标
VPC CIDR	本地
0.0.0.0/0	*internet-gateway-id*

以下是可用区 B 中与私有子网关联的路由表。第一个条目是本地路由；它使子网中的实例能够使用私有 IP 地址与 VPC 中的其他实例进行通信。第二个条目将所有其他子网流量发送到 NAT 网关。

目标位置	目标
VPC CIDR	本地
0.0.0.0/0	*nat-gateway-id*

有关更多信息，请参阅 the section called “管理子网路由表”。

测试公有 NAT 网关

创建 NAT 网关并更新路由表之后，您可以从私有子网中的实例对互联网 ping 一些远程地址以测试它是否可以连接到互联网。有关如何执行此操作的示例，请参阅 测试互联网连接。

如果能够连接到互联网，还可以测试互联网流量是否通过 NAT 网关进行路由：

- 跟踪来自私有子网中实例的流量的路由情况。为此，请从私有子网中的 Linux 实例运行 `traceroute` 命令。在输出中，应在一个跃点（通常是第一个跃点）中看到 NAT 网关的私有 IP 地址。
- 从私有子网中的实例连接第三方网站或工具时，查看该网站或工具显示的源 IP 地址。源 IP 地址应是 NAT 网关的弹性 IP 地址。

如果这些测试失败，请参阅 排查 NAT 网关的问题。

测试互联网连接

以下示例演示如何测试私有子网中的实例是否可以连接到互联网。

1. 在公有子网中启动实例（您使用此实例作为堡垒主机）。在启动向导中，确保选择一个 Amazon Linux AMI，并为实例分配公有 IP 地址。确保安全组规则允许来自本地网络的 IP 地址范围的入站 SSH 流量，以及发送到私有子网的 IP 地址范围的出站 SSH 流量（您也可以同时对入站和出站 SSH 流量使用 `0.0.0.0/0` 进行测试）。
2. 在您的私有子网中启动实例。在启动向导中，确保选择一个 Amazon Linux AMI。请勿向实例分配公有 IP 地址。应确保安全组规则允许来自在公有子网中启动的实例的私有 IP 地址的入站 SSH 流量以及所有出站 ICMP 流量。必须选择用于在公有子网中启动实例的相同密钥对。
3. 在本地计算机上配置 SSH 代理转发，并连接到公有子网中的堡垒主机。有关更多信息，请参阅 为 Linux 或 macOS 配置 SSH 代理转发 或 针对 Windows 配置 SSH 代理转发。
4. 在堡垒主机中，连接到私有子网中的实例，然后从私有子网中的实例测试 Internet 连接。有关更多信息，请参阅 测试 Internet 连接。

为 Linux 或 macOS 配置 SSH 代理转发

1. 在您的本地计算机上，将私有秘钥添加到身份验证代理。

   对于 Linux，请使用以下命令。

   ```
   ssh-add -c mykeypair.pem
   ```

   对于 macOS，请使用以下命令。

   ```
   ssh-add -K mykeypair.pem
   ```

2. 通过使用 -A 选项启用 SSH 代理转发来连接到公有子网中的实例，并使用该实例的公有地址，如以下示例所示。

```
ssh -A ec2-user@54.0.0.123
```

针对 Windows 配置 SSH 代理转发

您可以使用 Windows 中提供的 OpenSSH 客户端，也可以安装您的首选 SSH 客户端（例如 PuTTY）。

OpenSSH

按照 Getting started with OpenSSH for Windows 一文中的说明，安装适用于 Windows 的 OpenSSH。然后将密钥添加到身份验证代理。有关更多信息，请参阅 Key-based authentication in OpenSSH for Windows。

PuTTY

1. 如果尚未安装 Pageant，请从 PuTTY 下载页面下载并安装 Pageant。
2. 将您的私有密钥转换为 .ppk 格式。有关更多信息，请参阅《Amazon EC2 用户指南》中的使用 PuTTYgen 转换私有密钥。
3. 启动 Pageant，右键单击任务栏上的 Pageant 图标（可能已隐藏），并选择 Add Key（添加）。选择您创建的 .ppk 文件，输入密码（如果需要），然后选择 Open（打开）。
4. 启动 PuTTY 会话，并使用公有 IP 地址连接到公有子网中的实例。有关更多信息，请参阅使用 PuTTY 连接到 Linux 实例。在 Auth 类别中，确保选中 Allow agent forwarding（允许代理转发）选项，并将 Private key file for authentication（用于验证的私有密钥文件）框留空。

测试 Internet 连接

1. 从公有子网中的实例，使用私有 IP 地址连接到私有子网中的实例，如以下示例所示。

```
ssh ec2-user@10.0.1.123
```

2. 从私有实例，通过对启用了 ICMP 的网站运行 ping 命令来测试是否可以连接到 Internet。

```
ping ietf.org
```

```
PING ietf.org (4.31.198.44) 56(84) bytes of data.
64 bytes from mail.ietf.org (4.31.198.44): icmp_seq=1 ttl=47 time=86.0 ms
64 bytes from mail.ietf.org (4.31.198.44): icmp_seq=2 ttl=47 time=75.6 ms
...
```

按键盘上的 Ctrl+C 以取消 ping 命令。如果 ping 命令失败，请参阅 实例无法访问 Internet。

3. （可选）如果您不再需要实例，请将其终止。有关更多信息，请参阅《Amazon EC2 用户指南》中的 终止实例。

从允许列出的 IP 地址访问您的网络

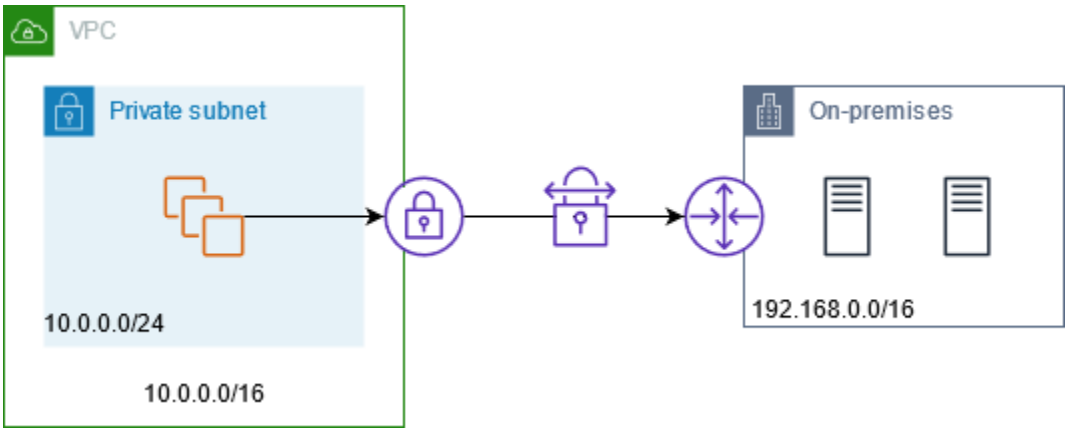
您可以通过私有 NAT 网关，使用允许列出的地址池来实现 VPC 与本地网络的通信。您可以通过私有 NAT 网关（具有来自允许列出的 IP 地址范围的 IP 地址）来路由发往本地网络的子网中的流量，而不是为每个实例分配一个来自允许列出的 IP 地址范围的单独 IP 地址。

内容

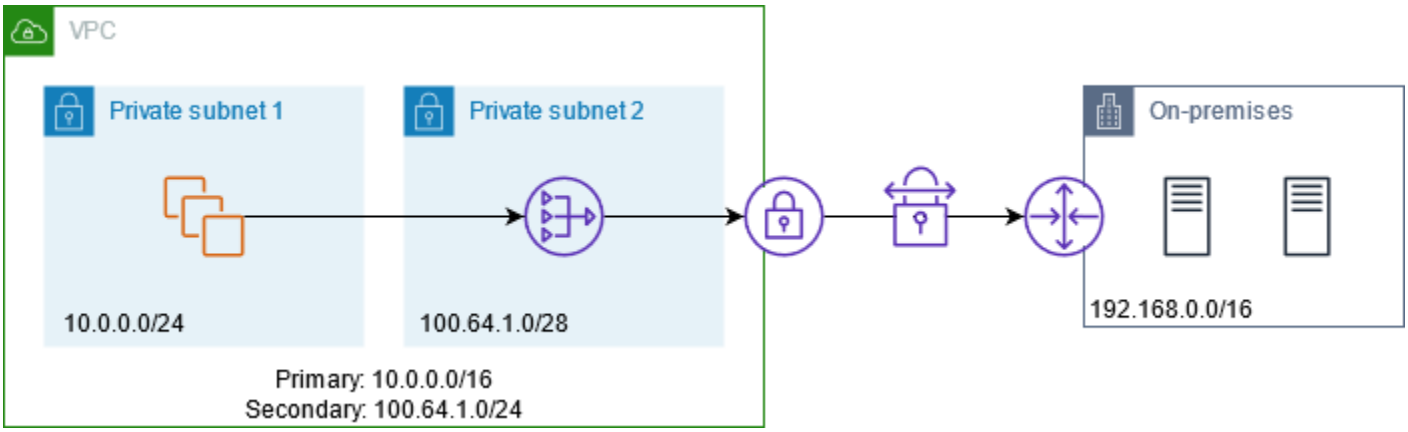
- 概览
- 资源
- 路由

概览

下图显示了实例如何通过 AWS VPN 访问本地资源。来自实例的流量通过 VPN 连接路由到虚拟私有网关，到达客户网关，然后到达本地网络中的目标。但是，假设目标仅允许来自特定 IP 地址范围（例如 100.64.1.0/28）的流量。这可防止来自这些实例的流量到达本地网络。



下表展示了此场景配置的主要组成部分。VPC 具有原始 IP 地址范围和允许的 IP 地址范围。VPC 有一个来自允许的 IP 地址范围的子网和一个私有 NAT 网关。来自实例的发往本地网络的流量会先发送到 NAT 网关，然后再路由到 VPN 连接。本地网络接收来自具有 NAT 网关源 IP 地址的实例的流量，该地址来自允许的 IP 地址范围。



资源

按如下方式创建或更新资源：

- 将允许的 IP 地址范围与 VPC 关联。
- 在 VPC 中从允许的 IP 地址范围创建子网。
- 在新子网中创建私有 NAT 网关。
- 使用实例更新子网的路由表，以将发往本地网络的流量发送到 NAT 网关。将路由添加到具有私有 NAT 网关的子网的路由表，该网关会将发往本地网络的流量发送到虚拟私有网关。

路由

以下是与第一个子网关联的路由表。每个 VPC CIDR 都有一个本地路由。本地路由使子网中的资源能够使用私有 IP 地址与 VPC 中的其他资源进行通信。第三个条目会将发往本地网络的流量发送到私有 NAT 网关。

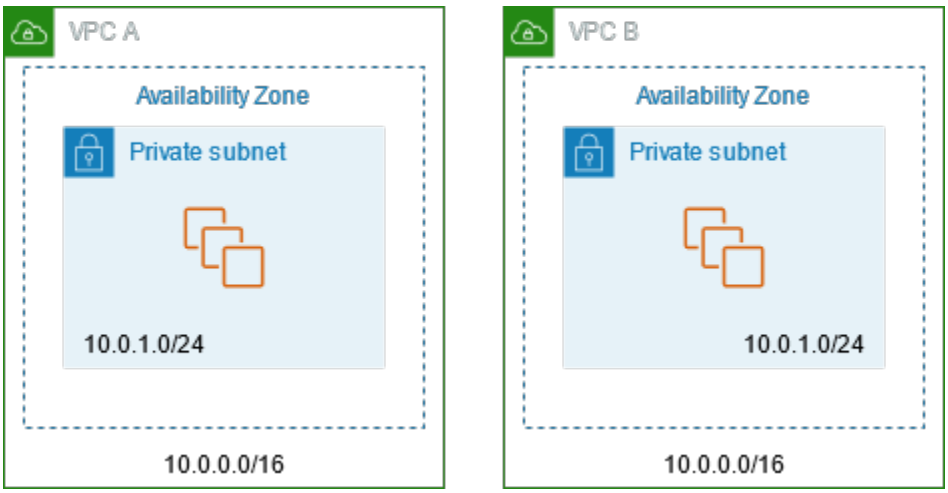
目标位置	目标
10.0.0.0/16	本地
100.64.1.0/24	本地
192.168.0.0/16	*nat-gateway-id*

以下是与第二个子网关联的路由表。每个 VPC CIDR 都有一个本地路由。本地路由使子网中的资源能够使用私有 IP 地址与 VPC 中的其他资源进行通信。第三个条目会将发往本地网络的流量发送到虚拟私有网关。

目标位置	目标
10.0.0.0/16	本地
100.64.1.0/24	本地
192.168.0.0/16	*vgw-id*

实现重叠网络之间的通信

即使网络具有重叠的 CIDR 范围，也可以使用私有 NAT 网关来启用网络之间的通信。例如，假设 VPC A 中的实例需要访问 VPC B 中的实例提供的服务。



内容

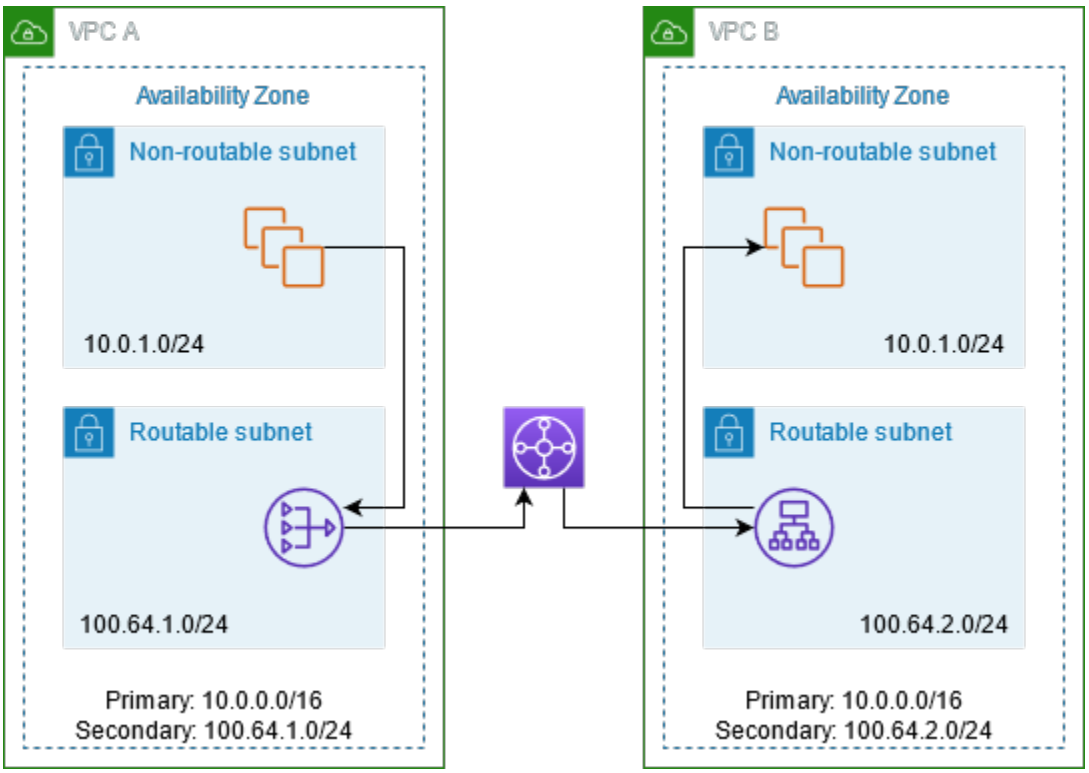
- 概览
- 资源
- 路由

概览

下表展示了此场景配置的主要组成部分。首先，您的 IP 管理团队需要确定哪些地址范围可以重叠（不可路由的地址范围），哪些地址范围不能重叠（可路由的地址范围）。IP 管理团队根据请求将可路由地址范围池中的地址范围分配给项目。

每个 VPC 都有其原始 IP 地址范围（不可路由）以及由 IP 管理团队分配给它的可路由 IP 地址范围。VPC A 有一个来自可路由范围的子网和一个私有 NAT 网关。私有 NAT 网关从其子网获取其 IP 地址。VPC B 有一个来自可路由范围的子网和一个应用程序负载均衡器。应用程序负载均衡器从其子网获取 IP 地址。

来自 VPC A 的不可路由子网中的实例的流量（将发往 VPC B 的不可路由子网中的实例）通过私有 NAT 网关发送，然后路由到中转网关。中转网关将流量发送到应用程序负载均衡器，后者将流量路由到 VPC B 的不可路由子网中的其中一个目标实例。从中转网关到应用程序负载均衡器的流量具有私有 NAT 网关的源 IP 地址。因此，来自负载均衡器的响应流量使用私有 NAT 网关的地址作为其目的。响应流量将发送到中转网关，然后路由到私有 NAT 网关，该网关会将目标转换为 VPC A 的不可路由子网中的实例。



资源

按如下方式创建或更新资源：

- 将分配的可路由 IP 地址范围与各自的 VPC 关联。

- 在 VPC A 中从可路由的 IP 地址范围创建子网，然后在此新子网中创建私有 NAT 网关。
- 在 VPC B 中从可路由的 IP 地址范围创建子网，然后在此新子网中创建应用程序负载均衡器。将不可路由子网中的实例注册到负载均衡器的目标组。
- 创建中转网关以连接 VPC。确保禁用路由传播。将每个 VPC 连接到中转网关时，请使用 VPC 的可路由地址范围。
- 更新 VPC A 中不可路由子网的路由表，以将发往 VPC B 的可路由地址范围的所有流量发送到私有 NAT 网关。更新 VPC A 中可路由子网的路由表，以将发往 VPC B 的可路由地址范围的所有流量发送到中转网关。
- 更新 VPC B 中可路由子网的路由表，以将发往 VPC A 的可路由地址范围的所有流量发送到中转网关。

路由

以下是 VPC A 中不可路由子网的路由表。

目标位置	目标
10.0.0.0/16	本地
100.64.1.0/24	本地
100.64.2.0/24	*nat-gateway-id*

以下是 VPC A 中可路由子网的路由表。

目标位置	目标
10.0.0.0/16	本地
100.64.1.0/24	本地
100.64.2.0/24	*transit-gateway-id*

以下是 VPC B 中不可路由子网的路由表。

目标位置	目标
10.0.0.0/16	本地
100.64.2.0/24	本地

以下是 VPC B 中可路由子网的路由表。

目标位置	目标
10.0.0.0/16	本地
100.64.2.0/24	本地
100.64.1.0/24	*transit-gateway-id*

以下是中转网关路由表。

CIDR	附件	路由类型
100.64.1.0/24	*VPC A ###*	静态
100.64.2.0/24	*VPC B ###*	静态

DNS64 和 NAT64

NAT 网关支持从 IPv6 到 IPv4 的网络地址转换，这通常称为 NAT64。NAT64 可以帮助您的 IPv6 AWS 资源与同一 VPC 或其他 VPC、本地网络或互联网中的 IPv4 资源进行通信。您可以在 Amazon Route 53 Resolver 上结合使用 NAT64 和 DNS64，也可以使用自己的 DNS64 服务器。

内容

- 什么是 DNS64？
- 什么是 NAT64？
- 配置 DNS64 和 NAT64

什么是 DNS64？

在 VPC 中运行的仅 IPv6 工作负载只能发送和接收 IPv6 网络数据包。如果没有 DNS64，则对仅 IPv4 服务的 DNS 查询将生成 IPv4 目标地址作为响应，而且仅 IPv6 服务无法与其进行通信。为了弥合这一通信缺口，您可以为子网启用 DNS64，它适用于该子网中的所有 AWS 资源。使用 DNS64，Amazon Route 53 Resolver 将查找所查询的服务的 DNS 记录，然后执行以下操作之一：

- 如果记录包含 IPv6 地址，则它将返回原始记录并建立连接，而不会通过 IPv6 进行任何转换。
- 如果 DNS 记录中没有与目标关联的 IPv6 地址，则 Route 53 Resolver 会在记录中的 IPv4 地址前面添加 RFC6052 (`64:ff9b::/96`) 中定义的已知 `/96` 前缀，以便合成一个地址。仅 IPv6 服务会将网络数据包发送到合成的 IPv6 地址。然后，您需要通过 NAT 网关路由此流量，该网关将对流量执行所需的转换，以允许子网中的 IPv6 服务访问该子网外部的 IPv4 服务。

您可以通过 AWS CLI 使用 modify-subnet-attribute 来启用或禁用子网上的 DNS64，也可以使用 VPC 控制台执行此操作，方法是选择子网，然后选择 Actions（操作）> Edit subnet settings（编辑子网设置）。

什么是 NAT64？

通过使用 NAT64，Amazon VPC 中的仅 IPv6 服务能够与同一 VPC（不同子网中）或已连接的 VPC、本地网络或互联网中的仅 IPv4 服务进行通信。

NAT64 将在现有的 NAT 网关或您创建的任何新 NAT 网关上自动可用。您无法启用或禁用此功能。NAT 网关所在的子网不需要是双堆栈子网即可使 NAT64 正常工作。

在您启用 DNS64 后，如果仅使用 IPv6 的服务通过 NAT 网关将网络数据包发送到合成的 IPv6 地址后，将发生以下情况：

- NAT 网关可通过 `64:ff9b::/96` 前缀识别出原始目标是 IPv4，并且会将 IPv6 数据包转换为 IPv4，方法是：
  - 将源 IPv6 替换为自己的私有 IP，该 IP 将被互联网网关转换为弹性 IP 地址。
  - 通过截断 `64:ff9b::/96` 前缀将目标 IPv6 转换为 IPv4。
- NAT 网关通过互联网网关、虚拟私有网关或转换网关将转换后的 IPv4 数据包发送到目标，然后启动连接。
- 仅 IPv4 主机将发回 IPv4 响应数据包。建立连接后，NAT 网关将接受来自外部主机的响应 IPv4 数据包。

- 响应 IPv4 数据包的目标是 NAT 网关，该网关将接收数据包，并通过将其 IP（目标 IP）替换为主机的 IPv6 地址并将 `64:ff9b::/96` 添加回源 IPv4 地址来取消 NAT。然后，数据包将按照本地路由流向主机。

通过这种方式，NAT 网关使子网中仅使用IPv6 的工作负载能够与子网外部的仅使用 IPv4 的服务进行通信。

配置 DNS64 和 NAT64

按照本节中的步骤配置 DNS64 和 NAT64，以便与仅 IPv4 服务进行通信。

内容

- 通过 AWS CLI 与互联网上的仅 IPv4 服务进行通信
- 在您的本地环境中启用与仅 IPv4 服务的通信

通过 AWS CLI 与互联网上的仅 IPv4 服务进行通信

如果您有一个包含仅 IPv6 工作负载的子网，它需要与其外部的仅 IPv4 服务进行通信，本示例将向您展示如何让这些仅 IPv6 服务与互联网上的仅 IPv4 服务进行通信。

您应该首先在公有子网（与包含仅 IPv6 工作负载的子网分开）中配置 NAT 网关。例如，包含 NAT 网关的子网应该具有指向互联网网关的 `0.0.0.0/0` 路由。

完成以下步骤以使这些仅 IPv6 的服务能够与互联网上的仅 IPv4 服务建立连接：

1. 将以下三个路由添加到包含仅 IPv6 工作负载的子网的路由表中：

   - 指向 NAT 网关的 IPv4 路由（如果有）。
   - 指向 NAT 网关的 `64:ff9b::/96` 路由。这将允许通过 NAT 网关路由发往仅 IPv4 服务的仅 IPv6 的工作负载中的流量。
   - 指向仅出口互联网网关（或互联网网关）的 IPv6 `::/0` 路由。

   请注意，将 `::/0` 指向互联网网关将允许外部 IPv6 主机（VPC 外）通过 IPv6 发起连接。

```
aws ec2 create-route --route-table-id rtb-34056078 --destination-cidr-block
0.0.0.0/0 --nat-gateway-id nat-05dba92075d71c408
```

```
aws ec2 create-route --route-table-id rtb-34056078 --destination-ipv6-cidr-block
64:ff9b::/96 --nat-gateway-id nat-05dba92075d71c408
```

```
aws ec2 create-route --route-table-id rtb-34056078 --destination-ipv6-cidr-block
::/0 --egress-only-internet-gateway-id eigw-c0a643a9
```

2. 在包含仅 IPv6 工作负载的子网中启用 DNS64 功能。

```
aws ec2 modify-subnet-attribute --subnet-id subnet-1a2b3c4d --enable-dns64
```

现在，私有子网中的资源可以通过互联网与 IPv4 和 IPv6 服务建立有状态的连接。正确配置安全组和 NACL，以允许至 64:ff9b::/96 流量的出口和入口流量。

在您的本地环境中启用与仅 IPv4 服务的通信

Amazon Route 53 Resolver 使您能够将 DNS 查询从 VPC 转发到本地网络，反之亦然。您可以通过以下步骤实现上述目的：

- 您可以在 VPC 中创建一个 Route 53 Resolver 出站端点，并为其分配您希望 Route 53 Resolver 通过其转发查询的 IPv4 地址。对于本地 DNS 解析程序，这些是 DNS 查询源自的 IP 地址，因此，应为 IPv4 地址。
- 您可以创建一个或多个规则，用于指定您希望 Route 53 Resolver 转发到本地解析程序的 DNS 查询的域名。此外，您还可以指定本地解析程序的 IPv4 地址。
- 既然您已设置 Route 53 Resolver 出站端点，那么，您需要在包含仅 IPv6 工作负载的子网上启用 DNS64，然后通过 NAT 网关路由发往本地网络的任何数据。

DNS64 如何适用于本地网络中的仅 IPv4 目标：

1. 将 IPv4 地址分配给 VPC 中的 Route 53 Resolver 出站端点。
2. 来自 IPv6 服务的 DNS 查询通过 IPv6 转至 Route 53 Resolver。Route 53 Resolver 将根据转发规则匹配查询，并获取本地解析程序的 IPv4 地址。
3. Route 53 Resolver 会将查询数据包从 IPv6 转换为 IPv4，并将其转发到出站端点。端点的每个 IP 地址代表一个 ENI，用于将请求转发到 DNS 解析程序的本地 IPv4 地址。
4. 本地解析程序通过 IPv4 将响应数据包从出站端点返回 Route 53 Resolver。
5. 假设查询是来自支持 DNS64 的子网进，则 Route 53 Resolver 将会执行以下两项操作：

a. 检查响应数据包的内容。如果记录中有 IPv6 地址，则它会将保持内容原样，但如果它只包含 IPv4 记录。它还会通过向 IPv4 地址追加 64:ff9b::/96 来合成 IPv6 记录。

b. 重新打包内容，并通过 IPv6 将其发送到 VPC 中的服务。

使用 Amazon CloudWatch 监控 NAT 网关

您可以使用 CloudWatch 监控 NAT 网关，该工具可从 NAT 网关中收集信息并创建可读的、近乎实时的指标。您可以使用该信息监控 NAT 网关并进行问题排查。这些指标让您可以了解 NAT 网关的运行状况和性能，从而能够密切监视网关的运行状况并快速排查任何问题。

CloudWatch 收集的 NAT 网关指标包含数据点，例如已处理的字节、数据包计数、连接计数和错误率。这使您能够全面了解流经自己 NAT 网关的流量，识别任何异常或瓶颈。CloudWatch 以 1 分钟为间隔传输此指标数据，可提供精确到分钟的 NAT 网关行为精细视图。

此外，CloudWatch 会将此 NAT 网关指标数据保留 15 个月，便于您分析一段时间内的趋势和模式。您可以使用这些历史数据来规划容量、优化性能以及了解 NAT 网关使用情况的长期演变。

要利用这些强大的监控功能，您可以针对自身特定需求创建自定义 CloudWatch 控制面板和警报。例如，可以设置警报，在 NAT 网关的出站数据传输超过特定阈值时发出通知，便于您主动解决潜在的带宽限制问题。

有关定价的更多信息，请参阅 Amazon CloudWatch 定价。

内容

- NAT 网关指标与维度
- 查看 NAT 网关 CloudWatch 指标
- 创建 CloudWatch 警报以监控 NAT 指标

NAT 网关指标与维度

以下指标可用于 NAT 网关。描述列包括每个指标的描述以及单位和统计数据。

指标	描述
ActiveConnectionCount	通过 NAT 网关激活的并发 TCP 连接的总数。 零值表示未通过 NAT 网关激活任何连接。

指标	描述
	单位：计数 Statistics：最有用的统计工具是 Max。
BytesInFromDestination	NAT 网关从目标接收的字节的数量。 如果 BytesOutToSource 的值小于 BytesInFromDestination 的值，则表示 NAT 网关处理期间可能存在数据丢失，或存在被 NAT 网关主动阻止的流量。 单位：字节 Statistics：最有用的统计工具是 Sum。
BytesInFromSource	NAT 网关从 VPC 中的客户端接收的字节的数量。 如果 BytesOutToDestination 的值小于 BytesInFromSource 的值，则 NAT 网关处理期间可能有数据丢失。 单位：字节 Statistics：最有用的统计工具是 Sum。
BytesOutToDestination	通过 NAT 网关发送到目标的字节的数量。 大于零的值指示有流量从 NAT 网关后面的客户端流向 Internet。如果 BytesOutToDestination 的值小于 BytesInFromSource 的值，则 NAT 网关处理期间可能有数据丢失。 单位：字节 Statistics：最有用的统计工具是 Sum。

指标	描述
BytesOutToSource	通过 NAT 网关发送到 VPC 中客户端的字节的数量。 大于零的值指示有流量从 Internet 流向 NAT 网关后面的客户端。如果 BytesOutToSource 的值小于 BytesInFromDestination 的值，则表示 NAT 网关处理期间可能存在数据丢失，或存在被 NAT 网关主动阻止的流量。 单位：字节 Statistics：最有用的统计工具是 Sum。
ConnectionAttemptCount	通过 NAT 网关尝试的连接次数。这仅包括初始 SYN。在某些情况下，由于 SYN 重新传输，ConnectionAttemptCount 可能低于 ConnectionEstablishedCount 。 如果 ConnectionEstablishedCount 的值小于 ConnectionAttemptCount 的值，则表示 NAT 网关后面的客户端已尝试为无响应的连接建立新连接。 单位：计数 Statistics：最有用的统计工具是 Sum。
ConnectionEstablishedCount	通过 NAT 网关建立的连接的数量。这包括 SYN 和 SYN 重新传输。 如果 ConnectionEstablishedCount 的值小于 ConnectionAttemptCount 的值，则表示 NAT 网关后面的客户端已尝试为无响应的连接建立新连接。 单位：计数 Statistics：最有用的统计工具是 Sum。

指标	描述
ErrorPortAllocation	NAT 网关无法分配源端口的次数。 大于零的值表示通过 NAT 网关打开的并发连接太多。 单位：计数 Statistics：最有用的统计工具是 Sum。
IdleTimeoutCount	从活动状态转换为空闲状态的连接的数量。如果活动连接未正常关闭并且前 350 秒内无活动，活动连接将转换为空闲状态。 大于零的值指示存在已变为空闲状态的连接。如果 IdleTimeoutCount 的值增加，则可能指示 NAT 网关后面的客户端正在重复使用过期连接。 单位：计数 Statistics：最有用的统计工具是 Sum。
PacketsDropCount	NAT 网关丢弃的数据包的数量。 要计算丢弃的数据包数量占数据包总流量的百分比，请使用以下公式：PacketsDropCount/(PacketsInFromSource+PacketsInFromDestination)*100 。如果该值超过 NAT 网关上总流量的 0.01%，则 Amazon VPC 服务可能存在问题。使用 AWS 服务运行状况控制面板来确定可能导致 NAT 网关丢包的服务问题。 单位：计数 Statistics：最有用的统计工具是 Sum。

指标	描述
PacketsInFromDestination	NAT 网关从目标接收的数据包的数量。 如果 PacketsOutToSource 的值小于 PacketsInFromDestination 的值，则表示 NAT 网关处理期间可能存在数据丢失，或存在被 NAT 网关主动阻止的流量。 单位：计数 Statistics：最有用的统计工具是 Sum。
PacketsInFromSource	NAT 网关从 VPC 中的客户端接收的数据包的数量。 如果 PacketsOutToDestination 的值小于 PacketsInFromSource 的值，则 NAT 网关处理期间可能有数据丢失。 单位：计数 Statistics：最有用的统计工具是 Sum。
PacketsOutToDestination	通过 NAT 网关发送到目标的数据包的数量。 大于零的值指示有流量从 NAT 网关后面的客户端流向 Internet。如果 PacketsOutToDestination 的值小于 PacketsInFromSource 的值，则 NAT 网关处理期间可能有数据丢失。 单位：计数 Statistics：最有用的统计工具是 Sum。

指标	描述
PacketsOutToSource	通过 NAT 网关发送到 VPC 中客户端的数据包的数量。 大于零的值指示有流量从 Internet 流向 NAT 网关后面的客户端。如果 PacketsOutToSource 的值小于 PacketsInFromDestination 的值，则表示 NAT 网关处理期间可能存在数据丢失，或存在被 NAT 网关主动阻止的流量。 单位：计数 Statistics：最有用的统计工具是 Sum。
PeakBytesPerSecond	该指标报告给定分钟内每秒的 10 秒字节最高平均值。 单位：计数 Statistics：最有用的统计工具是 Maximum。
PeakPacketsPerSecond	此指标每 10 秒计算一次平均数据包速率（每秒处理的数据包），持续 60 秒，然后报告六个速率中的最大值（最高平均数据包速率）。 单位：计数 Statistics：最有用的统计工具是 Maximum。

要筛选指标数据，请使用以下维度。

维度	描述
NatGatewayId	按 NAT 网关 ID 筛选指标数据。

查看 NAT 网关 CloudWatch 指标

NAT 网关指标按 1 分钟的时间间隔发送到 CloudWatch。指标的分组首先依据服务命名空间，然后依据每个命名空间内可能的维度组合。您可以按照以下方法查看 NAT 网关的各项指标。

使用 CloudWatch 控制台查看指标

1. 通过 https://console.aws.amazon.com/cloudwatch/ 打开 CloudWatch 控制台。
2. 在导航窗格中，依次选择 Metrics（指标）、All metrics（所有指标）。
3. 选择 NatGateway 指标命名空间。
4. 选择指标维度。

使用 AWS CLI 查看指标

在命令提示窗口中，使用以下命令可列出可用于 NAT 网关服务的指标。

```
aws cloudwatch list-metrics --namespace "AWS/NATGateway"
```

创建 CloudWatch 警报以监控 NAT 指标

您可以创建在警报改变状态时发送 Amazon SNS 消息的 CloudWatch 警报。警报会监控您指定的时间段内的某个指标。它将根据指标值在多个时间段内相对于给定阈值的情况向 Amazon SNS 主题发送通知。

例如，您可以创建警报来监控进入或离开 NAT 网关的流量。以下警报监控从您的 VPC 中的客户端通过 NAT 网关传到 Internet 的出站流量。如果在 15 分钟的时间段内字节数达到 500 万阈值，它将发送通知。

创建通过 NAT 网关的出站流量的警报

1. 访问 https://console.aws.amazon.com/cloudwatch/，打开 CloudWatch 控制台。
2. 在导航窗格中，依次选择 Alarms（警报）和 All alarms（所有警报）。
3. 选择创建警报。
4. 选择选择指标。
5. 选择 NatGateway 指标命名空间，然后选择指标维度。访问指标后，请选中 NAT 网关 BytesOutToDestination 指标旁边的复选框，然后选择 Select metric（选择指标）。
6. 按如下所示配置警报，然后选择 Next（下一步）：

- 对于 Statistic（统计数据），选择 Sum（总计）。
- 对于 Period（周期），选择 15 minutes（15 分钟）。
- 对于 Whenever（每当），选择 Greater/Equal（大于/等于，>=），然后输入 5000000 作为阈值。

7. 对于 Notification（通知），选择现有的 SNS 主题，或选择 Create new topic（新建主题）创建一个新主题。选择 Next（下一步）。
8. 输入警报的名称和描述，然后选择 Next（下一步）。
9. 配置完警报后，选择 Create alarm（创建警报）。

再给一个示例，您可以创建一个警报来监控端口分配错误，并且在该值在三个连续 5 分钟的时间段内大于零（0）时发送通知。

创建警报以监控端口分配错误

1. 访问 https://console.aws.amazon.com/cloudwatch/，打开 CloudWatch 控制台。
2. 在导航窗格中，依次选择 Alarms（警报）和 All alarms（所有警报）。
3. 选择创建警报。
4. 选择选择指标。
5. 选择 NatGateway 指标命名空间，然后选择指标维度。访问指标后，请选中 NAT 网关 ErrorPortAllocation 指标旁边的复选框，然后选择 Select metric（选择指标）。
6. 按如下所示配置警报，然后选择 Next（下一步）：

   - 对于 Statistic（统计数据），选择 Maximum（最大）。
   - 对于 Period（周期），选择 5 minutes（5 分钟）。
   - 对于 Whenever（每当），选择 Greater（大于），然后输入 0 作为阈值。
   - 对于 Additional configuration（其他配置）、Datapoints to alarm（警报的数据点数），输入 3。

7. 对于 Notification（通知），选择现有的 SNS 主题，或选择 Create new topic（新建主题）创建一个新主题。选择 Next（下一步）。
8. 输入警报的名称和描述，然后选择 Next（下一步）。
9. 配置完警报后，选择 Create alarm（创建警报）。

有关更多信息，请参阅 Amazon CloudWatch 用户指南中的使用 Amazon CloudWatch 告警。

排查 NAT 网关的问题

以下主题可帮助您排查在创建或使用 NAT 网关时可能遇到的常见问题。

问题

- NAT 网关创建失败
- NAT 网关配额
- 弹性 IP 地址配额
- 不支持可用区
- NAT 网关不再可见
- NAT 网关不响应 Ping 命令
- 实例无法访问 Internet
- 到目标的 TCP 连接失败
- 追踪路由输出未显示 NAT 网关私有 IP 地址
- Internet 连接在 350 秒后中断
- 无法建立 IPsec 连接
- 无法发起更多连接

NAT 网关创建失败

问题

您创建一个 NAT 网关，但它进入的状态为 `Failed`。

> **Note**
>
> 出现故障的 NAT 网关会被自动删除，通常在大约一小时内。

原因

创建 NAT 网关时出错。返回的状态消息提供了出现此错误的原因。

解决方案

要查看错误消息，请打开 Amazon VPC 控制台，然后选择 NAT Gateways（NAT 网关）。选择 NAT 网关对应的单选按钮，然后在 Details（详细信息）选项卡上找到 State message（状态消息）。

下表列出 Amazon VPC 控制台中指示的可能的失败原因。执行所示任何纠正步骤之后，您可以再次尝试创建 NAT 网关。

显示的错误	原因	解决方案
子网没有足够的空闲地址来创建此 NAT 网关	指定的子网没有任何空闲的私有 IP 地址。NAT 网关需要从子网范围分配了私有 IP 地址的网络接口。	检查子网中可用的 IP 地址数，方法是在 Amazon VPC 控制台中前往 Subnets (子网) 页面。您可以在子网的详细信息创窗格中查看 Available IP。要在子网中创建空闲的 IP 地址，可以删除未使用的网络接口，或终止不需要的实例。
网络 vpc-xxxxxxxx 未连接任何互联网网关	必须在具有 Internet 网关的 VPC 中创建 NAT 网关。	创建 Internet 网关，并将其连接到您的 VPC。有关更多信息，请参阅 向子网添加互联网访问权限。
弹性 IP 地址 eipalloc-xxxxxxxx 已关联	指定的弹性 IP 地址已与其他资源关联，无法与 NAT 网关相关联。	检查哪个资源与弹性 IP 地址相关联。前往 Amazon VPC 控制台中的 Elastic IPs (弹性 IP) 页面，并查看为实例 ID 或网络接口 ID 指定的值。如果该资源不需要该弹性 IP 地址，则可以解除两者的关联。或者，也可以向您的账户分配新的弹性 IP 地址。有关更多信息，请参阅 开始使用弹性 IP 地址。

NAT 网关配额

您在尝试创建 NAT 网关时收到以下错误。

```
Performing this operation would exceed the limit of 5 NAT gateways
```

原因

您已达到该可用区域的 NAT 网关数量的配额。

解决方案

如果您已达到此 NAT 网关在账户中的配额，则可以执行以下操作之一：

- 使用 Service Quotas 控制台请求增加每个可用区域的 NAT 网关配额。
- 检查 NAT 网关的状态。Pending、Available 或 Deleting 状态的网关就占用限额。如果您最近删除了 NAT 网关，请等待几分钟，以便状态从 Deleting 变为 Deleted。然后尝试新建一个 NAT 网关。
- 如果您在特定可用区中不需要 NAT 网关，请尝试在未达到配额的可用区中创建 NAT 网关。

有关更多信息，请参阅 Amazon VPC 配额。

弹性 IP 地址配额

问题

您在尝试为公用 NAT 网关分配弹性 IP 地址时收到以下错误。

```
The maximum number of addresses has been reached.
```

原因

您已达到该区域账户的弹性 IP 地址数量的配额。

解决方案

如果您的弹性 IP 地址数已达到配额，则可以取消弹性 IP 地址与其他资源的关联，或者，您可以使用 Service Quotas 控制台请求增加弹性 IP 配额。

不支持可用区

问题

您在尝试创建 NAT 网关时收到以下错误：NotAvailableInZone

原因

您可能会尝试在受约束的可用区（即我们的扩展能力受约束的区域）中创建 NAT 网关。

解决方案

我们无法在这些可用区中支持 NAT 网关。您可以在不同可用区中创建 NAT 网关并将它用于受约束区域中的私有子网。您还可以将资源移动到不受约束的可用区，以便您的资源和 NAT 网关处于同一区中。

NAT 网关不再可见

问题

您创建了一个 NAT 网关，但它在 Amazon VPC 控制台中不可见。

原因

创建 NAT 网关期间可能出错，创建失败。状态为 `Failed` 的 NAT 网关在 Amazon VPC 控制台中保持可见大约一小时。一个小时之后会被自动删除。

解决方案

查看 NAT 网关创建失败 中的信息，然后尝试创建新 NAT 网关。

NAT 网关不响应 Ping 命令

问题

如果您尝试从互联网（例如从家庭计算机）或从 VPC 中的任何实例对 NAT 网关的弹性 IP 地址或私有 IP 地址执行 ping 操作，则不会收到响应。

原因

NAT 网关仅从私有子网中的实例向 Internet 传输流量。

解决方案

要测试 NAT 网关是否正常运行，请参阅 测试公有 NAT 网关。

实例无法访问 Internet

问题

您创建了一个公有 NAT 网关并按照步骤进行了测试，但 `ping` 命令失败，或者您私有子网中的实例无法访问互联网。

原因

出现此问题的原因可能是以下原因之一：

- NAT 网关尚未准备好提供流量。
- 您的路由表未得到正确配置。
- 您的安全组或网络 ACL 阻止入站或出站流量。
- 您使用的是不受支持的协议。

解决方案

检查以下信息：

- 检查 NAT 网关是否处于 `Available` 状态。在 Amazon VPC 控制台中，转到 NAT 网关页面，然后在详细信息窗格中查看状态信息。如果 NAT 网关处于失败状态，则表示在创建它时可能发生了错误。有关更多信息，请参阅 NAT 网关创建失败。
- 检查您是否正确配置了路由表：
  - NAT 网关所处的公有子网必须具有将 Internet 流量路由到 Internet 网关的路由表。
  - 实例所处的私有子网必须具有将 Internet 流量路由到 NAT 网关的路由表。
  - 检查是否没有其他路由表条目将全部或部分 Internet 流量路由到其他设备而不是 NAT 网关。
- 确保私有实例的安全组规则允许出站 Internet 流量。要使 `ping` 命令正常运行，这些规则还必须允许出站 ICMP 流量。

  NAT 网关本身允许所有出站流量以及响应出站请求时收到的流量 (因此它是有状态的)。
- 确保与私有子网和公有子网关联的网络 ACL 没有阻止入站或出站 Internet 流量的规则。要使 `ping` 命令正常运行，这些规则还必须允许入站和出站 ICMP 流量。

  可以启用流日志以帮助诊断由于网络 ACL 或安全组规则而中断的连接。有关更多信息，请参阅 使用 VPC 流日志记录 IP 流量。
- 如果使用 `ping` 命令，请确保在对启用了 ICMP 的主机执行 ping 操作。如果未启用 ICMP，您不会收到应答数据包。要对此进行测试，请从您自己计算机上的命令行终端执行相同的 `ping` 命令。
- 检查实例是否能够对其他资源成功执行 ping 操作，例如私有子网中的其他实例 (假设安全组规则允许这样做)。
- 确保您的连接仅使用 TCP、UDP 或 ICMP 协议。

到目标的 TCP 连接失败

问题

在通过 NAT 网关从私有子网中的实例连接到特定目标时，有些 TCP 连接会成功，但也有些连接会失败或超时。

原因

出现此问题的原因可能是以下原因之一：

- 目标终端节点正在使用分段 TCP 数据包进行响应。NAT 网关不支持 TCP 或 ICMP 的 IP 碎片。有关更多信息，请参阅 比较 NAT 网关和 NAT 实例。
- 远程服务器上启用了 `tcp_tw_recycle` 选项，当 NAT 设备后有多个连接时，启用该选项会导致问题。

解决方案

通过执行以下操作，验证您尝试连接的终端节点是否正在使用分段 TCP 数据包进行响应：

1. 使用具有公共 IP 地址的公有子网中的实例来触发足够大的响应，以产生来自特定终端节点的分段。
2. 使用 `tcpdump` 实用工具验证终端节点是否将发送分段数据包。

   > **Important**
   >
   > 您必须使用公有子网中的实例来执行这些检查。您不能使用原始连接失败的实例，或者 NAT 网关或 NAT 实例后面的私有子网中的实例。

   发送或接收大型 ICMP 数据包的诊断工具将报告数据包丢失。例如，命令 `ping -s 10000 example.com` 将不会在 NAT 网关后面工作。
3. 如果终端节点发送分段 TCP 数据包，则可使用 NAT 实例代替 NAT 网关。

如果您有权访问远程服务器，则可以通过执行以下操作来验证是否已启用 `tcp_tw_recycle` 选项：

1. 在服务器上运行以下命令：

   ```
   cat /proc/sys/net/ipv4/tcp_tw_recycle
   ```

   如果输出为 `1`，则表明已启用 `tcp_tw_recycle` 选项。
2. 如果已启用 `tcp_tw_recycle` 选项，建议将其禁用。如果您需要重用连接，则 `tcp_tw_reuse` 是一个较为安全的选项。

如果您无权访问远程服务器，则可以通过临时禁用私有子网中的实例上的 `tcp_timestamps` 选项来进行测试。然后重新连接到远程服务器。如果连接成功，则上次连接失败的原因很可能是在远程服务器上启用了 `tcp_tw_recycle` 选项。如果可能，请与远程服务器的拥有者联系，以验证是否已启用此选项，如已启用，则请求将其禁用。

追踪路由输出未显示 NAT 网关私有 IP 地址

问题

您的实例可以访问 Internet，但是当您执行 `traceroute` 命令时，输出未显示 NAT 网关的私有 IP 地址。

原因

您的实例在使用其他网关（例如互联网网关）访问互联网。

解决方案

在实例所处的子网的路由表中，检查以下信息：

- 确保存在将 Internet 流量发送到 NAT 网关的路由。
- 确保没有其他特定路由将 Internet 流量发送到其他设备 (如虚拟私有网关或 Internet 网关)。

Internet 连接在 350 秒后中断

问题

您的实例可以访问互联网，但连接在 350 秒后断开。

原因

如果使用 NAT 网关的连接空闲 350 秒或更长时间，则连接会超时。

如果连接超时，NAT 网关向 NAT 网关后方的任何资源返回 RST 数据包，尝试继续进行连接 (它不发送 FIN 数据包)。

解决方案

要防止连接中断，您可以通过该连接发起更多流量。或者，您也可以在实例上启用值小于 350 秒的 TCP keepalive。

无法建立 IPsec 连接

问题

您无法与目标建立 IPsec 连接。

原因

NAT 网关当前不支持 IPsec 协议。

解决方案

您可以使用 NAT 遍历 (NAT-T) 将 IPsec 流量封装在 UDP（NAT 网关的支持协议）中。请确保您已测试您的 NAT-T 和 IPsec 配置，以验证您没有丢弃 IPsec 流量。

无法发起更多连接

问题

您有通过 NAT 网关与目标建立的现有连接，但您无法建立更多连接。

原因

您可能已达到单个 NAT 网关的并发连接数限制。有关更多信息，请参阅 NAT 网关基础知识。如果私有子网中的实例创建了大量连接，则您可能会达到该限制。

解决方案

请执行以下操作之一：

- 对每个可用区创建一个 NAT 网关，并在这些区域间分布客户端。
- 在公有子网中创建更多 NAT 网关并将客户端拆分到多个私有子网中 (各自具有指向不同 NAT 网关的路由)。
- 限制客户端可对目的地创建的连接数。
- 使用 CloudWatch 中的 `IdleTimeoutCount` 指标可监控空闲连接的增加。关闭空闲连接以释放容量。
- 创建具有多个 IP 地址的 NAT 网关或向现有的 NAT 网关添加辅助 IP 地址。每个新的 IPv4 地址最多可以支持 55,000 个并发连接。有关更多信息，请参阅创建 NAT 网关或编辑辅助 IP 地址关联。

适用于 NAT 网关的定价

当您预置 NAT 网关时，NAT 网关可用的每个小时及其处理的每个 GB 数据都需支付费用。有关更多信息，请参阅 Amazon VPC 定价。

以下策略可帮助您降低 NAT 网关的数据传输费用：

- 如果您的 AWS 资源会跨可用区发送或接收大量流量，则请确保资源与 NAT 网关位于同一可用区，或者在与资源相同的每个可用区中创建一个 NAT 网关。
- 如果通过 NAT 网关的大多数流量是到支持接口端点或网关端点的 AWS 服务，则请考虑为这些服务创建接口端点或网关端点。有关潜在成本节约的更多信息，请参阅 AWS PrivateLink 定价。

NAT 实例

NAT 实例提供网络地址转换（NAT）。您可以使用 NAT 实例允许私有子网中的资源与虚拟私有云（VPC）之外的目标通信，例如互联网或本地网络。私有子网中的资源可以向互联网发起出站 IPv4 流量，但它们无法接收在互联网上发起的入站流量。

> **Important**
>
> NAT AMI 基于 Amazon Linux AMI 的最新版本（2018.03）构建，该版本于 2020 年 12 月 31 日终止标准支持，并于 2023 年 12 月 31 日终止维护支持。有关更多信息，请参阅以下博客文章：Amazon Linux AMI 生命周期终止。
> 如果您使用现有的 NAT AMI，AWS建议您迁移到 NAT 网关。NAT 网关可提供更高的可用性、更高的带宽，并且所需的管理工作更少。有关更多信息，请参阅 比较 NAT 网关和 NAT 实例。如果 NAT 实例比 NAT 网关更适合您的使用案例，您可以从当前版本的 Amazon Linux 创建自己的 NAT AMI，如 the section called “3. 创建 NAT AMI” 中所述。

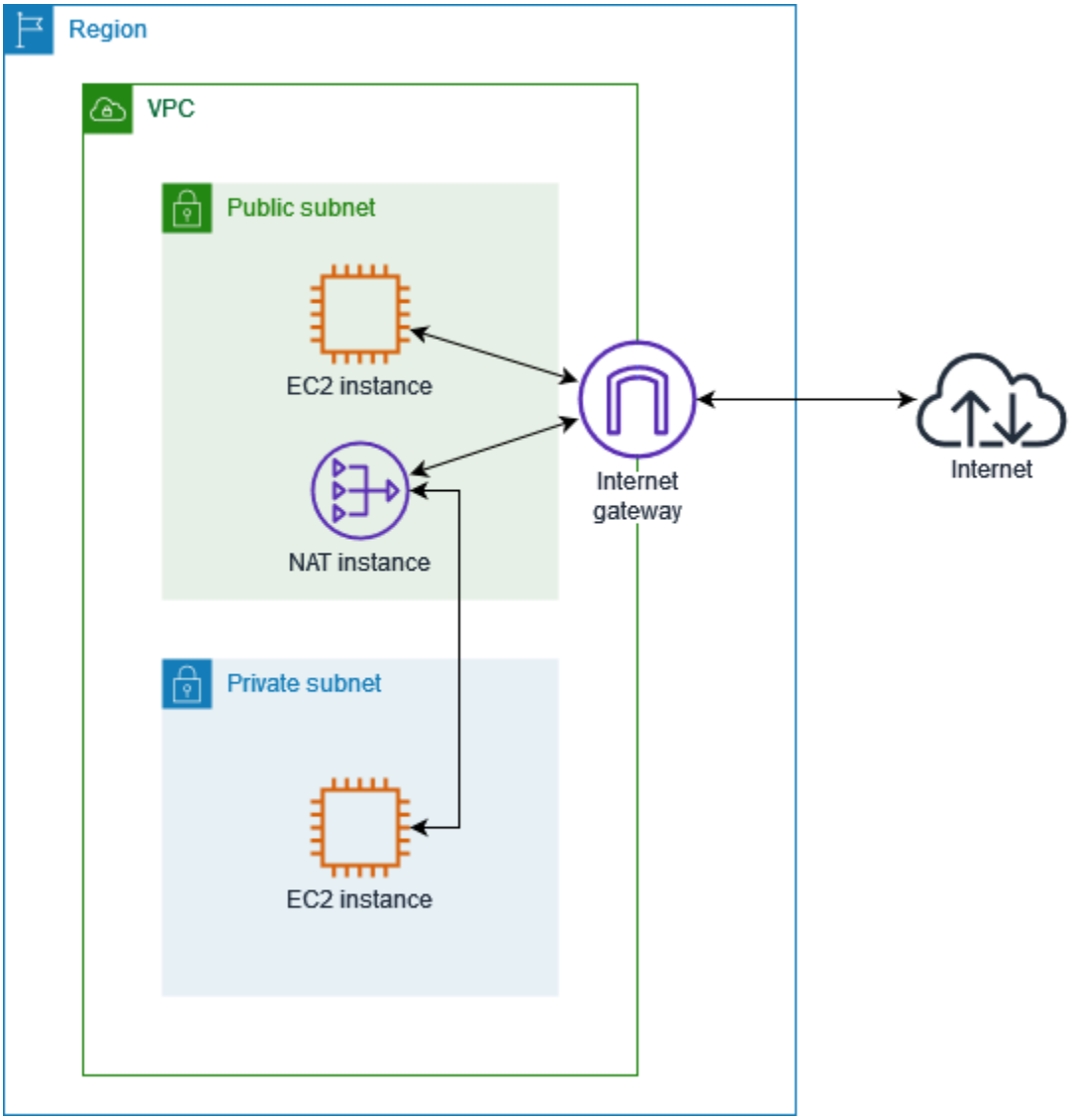
内容

- NAT 实例基础知识
- 允许私有资源在 VPC 外部进行通信

NAT 实例基础知识

下图展示了 NAT 实例的基本信息。路由表与私有子网关联，并将来自私有子网中实例的互联网流量发送到公有子网中的 NAT 实例。然后，NAT 实例再将流量发送到互联网网关。流量由 NAT 实例的公有 IP 地址产生。NAT 实例为响应指定了一个较高的端口号；响应返回后，NAT 实例会根据响应的端口号将其发送给私有子网中的相应实例。

NAT 实例必须具有互联网访问权限，因此它必须位于公有子网（路由表中包含通往互联网网关的路由的子网）中，并且必须具有公有 IP 地址或弹性 IP 地址。



要开始使用 NAT 实例，请创建 NAT AMI，为 NAT 实例创建安全组，然后将 NAT 实例启动到您的 VPC 中。

您的 NAT 实例配额取决于您在该区域的实例配额。有关更多信息，请参阅 AWS 一般参考 中的 Amazon EC2 服务配额。

允许私有资源在 VPC 外部进行通信

本节旨在介绍如何创建并使用 NAT 实例，让私有子网中的资源能够在虚拟私有云之外进行通信。

任务

- 1. 为 NAT 实例创建 VPC
- 2. 为 NAT 实例创建安全组

- 3. 创建 NAT AMI
- 4. 启动 NAT 实例
- 5. 禁用源/目标检查
- 6. 更新路由表
- 7. 测试您的 NAT 实例

1. 为 NAT 实例创建 VPC

使用以下过程创建具有公有和私有子网的 VPC。

创建 VPC

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 选择创建 VPC。
3. 对于 Resources to create（要创建的资源），选择 VPC and more（VPC 等）。
4. 对于 Name tag auto-generation（名称标签自动生成），为 VPC 输入名称。
5. 若要配置子网，请执行以下操作：
   a. 对于 Number of Availability Zones（可用区域数量），根据您的需求选择 1 或 2。
   b. 对于 Number of public subnets（公有子网数量），确保每个可用区有一个公有子网。
   c. 对于 Number of private subnets（私有子网数量），确保每个可用区有一个私有子网。
6. 选择创建 VPC。

2. 为 NAT 实例创建安全组

使用下表中描述的规则创建安全组。这些规则允许您的 NAT 实例从私有子网中的实例接收互联网范围流量以及来自您的网络的 SSH 流量。NAT 实例也可以向 Internet 发送流量，即允许私有子网中的实例获取软件更新。

以下是推荐的入站规则。

来源	协议	端口范围	注释
#### CIDR	TCP	80	允许来自私有子网服务器的入站 HTTP 数据流

来源	协议	端口范围	注释
#### CIDR	TCP	443	允许来自私有子网服务器的入站 HTTPS 数据流
####### IP ####	TCP	22	允许从您的网络到 NAT 实例的入站 SSH 访问（通过互联网网关）

以下是推荐的出站规则。

目标位置	协议	端口范围	注释
0.0.0.0/0	TCP	80	允许对 Internet 进行出站 HTTP 访问
0.0.0.0/0	TCP	443	允许对 Internet 进行出站 HTTPS 访问

创建安全组

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Security Groups（安全组）。
3. 选择Create security group（创建安全组）。
4. 输入安全组的名称和描述。
5. 对于 VPC，选择用于 NAT 实例的 VPC 的 ID。
6. 在入站规则下添加入站流量规则，如下所示：
   a. 选择 添加规则。在类型中选择 HTTP，并在源中输入私有子网的 IP 地址范围。
   b. 选择 添加规则。在类型中选择 HTTPS，并在源中输入私有子网的 IP 地址范围。
   c. 选择 添加规则。在类型中选择 SSH，并在源中输入网络的 IP 地址范围。
7. 在出站规则下添加出站流量规则，如下所示：
   a. 选择 添加规则。在类型中选择 HTTP，并在目的地中输入 0.0.0.0/0。
   b. 选择 添加规则。在类型中选择 HTTPS，并在目的地中输入 0.0.0.0/0。
8. 选择创建安全组。

有关更多信息，请参阅 安全组。

3. 创建 NAT AMI

NAT AMI 被配置为在 EC2 实例上运行 NAT。您必须先创建 NAT AMI，然后使用您的 NAT AMI 启动 NAT 实例。

如果您计划为 NAT AMI 使用 Amazon Linux 以外的操作系统，请参阅该操作系统的文档了解如何配置 NAT。请务必保存这些设置，以便其在实例重启后依然保持不变。

为 Amazon Linux 创建 NAT AMI

1. 启动运行 AL2023 或 Amazon Linux 2 的 EC2 实例。请务必指定为 NAT 实例创建的安全组。
2. 连接到实例并在实例上运行以下命令以启用 iptables。

```
sudo yum install iptables-services -y
sudo systemctl enable iptables
sudo systemctl start iptables
```

3. 在实例上执行以下操作以启用 IP 转发，使其在重启后仍然存在：
   a. 使用文本编辑器（例如 nano 或 vim）创建以下配置文件：`/etc/sysctl.d/custom-ip-forwarding.conf`。
   b. 将以下行添加到配置文件。

```
net.ipv4.ip_forward=1
```

   c. 保存配置文件，退出文本编辑器。
   d. 运行以下命令以应用配置文件。

```
sudo sysctl -p /etc/sysctl.d/custom-ip-forwarding.conf
```

4. 在实例上运行以下命令，并记下主网络接口的名称。您在下一步中需要此信息。

```
netstat -i
```

   在以下示例输出中，`docker0` 是 docker 创建的网络接口，`eth0` 是主网络接口，`lo` 是环回接口。

```
Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
```

```
docker0   1500        0      0      0 0             0      0      0      0 BMU
eth0      9001  7276052      0      0 0       5364991      0      0      0 BMRU
lo       65536   538857      0      0 0        538857      0      0      0 LRU
```

在以下示例输出中，主网络接口为 enX0。

```
Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
enX0      9001     1076      0      0 0          1247      0      0      0 BMRU
lo       65536       24      0      0 0            24      0      0      0 LRU
```

在以下示例输出中，主网络接口为 ens5。

```
Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
ens5      9001    14036      0      0 0          2116      0      0      0 BMRU
lo       65536       12      0      0 0            12      0      0      0 LRU
```

5. 在实例上运行以下命令以配置 NAT。如果主网络接口不是 eth0，请将 *eth0* 替换为您在上一步中记下的主网络接口。

```
sudo /sbin/iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
sudo /sbin/iptables -F FORWARD
sudo service iptables save
```

6. 然后，从 EC2 实例创建 NAT AMI。有关更多信息，请参阅《Amazon EC2 用户指南》中的从实例创建 Linux AMI。

4. 启动 NAT 实例

按照以下步骤使用您创建的 VPC、安全组和 NAT AMI 启动 NAT 实例。

启动 NAT 实例

1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 在控制面板上，选择启动实例。
3. 对于名称，输入您的 NAT 实例名称。
4. 对于应用程序和操作系统映像，选择您的 NAT AMI（选择浏览更多 AMI、我的 AMI）。
5. 对于实例类型，选择选择一个实例类型，以提供 NAT 实例所需的计算、内存和存储资源。
6. （可选）对于密钥对，选择一个现有密钥对或选择创建新密钥对。

7. 对于 Network settings（网络设置），执行以下操作：
   a. 选择编辑。
   b. 对于 VPC，选择已创建的 VPC。
   c. 对于子网，选择您创建的公有子网。
   d. 对于 Auto-assign public IP（自动分配公有 IP），选择 Enable（启用）。或者在启动 NAT 实例后，分配一个弹性 IP 地址并将其分配给 NAT 实例。
   e. 对于防火墙，选择选择现有安全组，然后选择您创建的安全组。
8. 选择启动实例。选择实例 ID 以打开实例详细信息页面。等待实例状态变为正在运行，并等待状态检查成功。
9. 禁用 NAT 实例的源/目的地检查（参阅 5. 禁用源/目标检查）。
10. 更新路由表以将流量发送至 NAT 实例（参阅 6. 更新路由表）。

5. 禁用源/目标检查

每项 EC2 实例都会默认执行源/目标检查。这意味着实例必须为其发送或接收的数据流的源头或目标。但是，NAT 实例必须能够在源或目标并非其本身时发送和接收数据流。因此，您必须禁用 NAT 实例的源/目标检查。

禁用源/目的地检查

1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 在导航窗格中，选择实例。
3. 选择 NAT 实例。
4. 依次选择操作、联网、更改源/目的地检查。
5. 对于源/目的地检查，请选择停止。
6. 选择保存。
7. 如果 NAT 实例有辅助网络接口，请从联网选项卡上的网络接口选择接口。选择接口 ID 以转至网络接口页面。选择操作、更改源/目标检查，清除启用，然后选择保存。

6. 更新路由表

私有子网的路由表必须有一个将互联网流量发送到 NAT 实例的路由。

更新路由表

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Route tables（路由表）。
3. 选择私有子网的路由表。
4. 在路由选项卡中选择编辑路由，然后选择添加路由。
5. 在目的地中输入 0.0.0.0/0，并在目标中输入 NAT 实例的实例 ID。
6. 选择 Save changes（保存更改）。

有关更多信息，请参阅 配置路由表。

7. 测试您的 NAT 实例

启动 NAT 实例并完成以上配置步骤之后，您可以测试私有子网中的实例是否可以通过将 NAT 实例用作堡垒机服务器来访问互联网。

任务

- 步骤 1：更新 NAT 实例安全组
- 步骤 2：在私有子网中启动测试实例
- 步骤 3：Ping 启用了 ICMP 的网站
- 步骤 4：清除

步骤 1：更新 NAT 实例安全组

如需允许私有子网中的实例向 NAT 实例发送 ping 流量，请添加一个规则来允许入站和出站 ICMP 流量。要允许 NAT 实例用作堡垒机服务器，请添加一个规则以允许出站 SSH 流量流向私有子网。

更新 NAT 实例安全组

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Security Groups（安全组）。
3. 选择与 NAT 实例关联的安全组的复选框。
4. 在 Inbound Rules (入站规则) 选项卡上，选择 Edit inbound rules (编辑入站规则)。
5. 选择 Add rule。对于类型，选择所有 ICMP - IPv4。对于源，选择自定义，输入您的私有子网的 IP 地址范围。选择保存规则。

6. 在出站规则选项卡上，选择编辑出站规则。
7. 选择 Add rule。对于类型，选择 SSH。对于目的地，选择自定义，并输入您的私有子网的 IP 地址范围。
8. 选择 Add rule。对于类型，选择所有 ICMP - IPv4。为 Destination (目的地) 选择 Anywhere - IPv4 (任何位置 – IPv4)。选择保存规则。

步骤 2：在私有子网中启动测试实例

在您的私有子网中启动实例。您必须允许从 NAT 实例进行 SSH 访问，并且必须使用用于 NAT 实例的相同密钥对。

在私有子网中启动测试实例

1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 在控制面板上，选择启动实例。
3. 选择您的私有子网。
4. 请勿向此实例分配公有 IP 地址。
5. 确保此实例的安全组允许来自您的 NAT 实例或公有子网的 IP 地址范围的入站 SSH 访问以及出站 ICMP 流量。
6. 选择用于 NAT 实例的相同密钥对。

步骤 3：Ping 启用了 ICMP 的网站

要验证私有子网中的测试实例是否可以使用 NAT 实例与互联网通信，请运行 ping 命令。

从您的私有实例测试互联网连接

1. 在本地计算机上，配置 SSH 代理转发，以便您可以将 NAT 实例用作堡垒机服务器。

   Linux and macOS

   ```
   ssh-add key.pem
   ```

   Windows

   下载并安装 Pageant（如果尚未安装）。

   使用 PuTTYgen 转换私有密钥。

启动 Pageant，右键单击任务栏上的 Pageant 图标（可能已隐藏），并选择添加密钥。选择您创建的 .ppk 文件，输入密码（如果需要），然后选择打开。

2. 从本地计算机连接到您的 NAT 实例。

   Linux and macOS

   ```
   ssh -A ec2-user@nat-instance-public-ip-address
   ```

   Windows

   使用 PuTTY 连接到您的 NAT 实例。对于身份验证，您必须选择允许代理转发，然后将用于身份验证的私有密钥文件留空。

3. 在 NAT 实例中，运行 ping 命令，指定启用 ICMP 的网站。

   ```
   [ec2-user@ip-10-0-4-184]$ ping ietf.org
   ```

   要确认您的 NAT 实例可以访问互联网，请验证您是否收到了如下输出，然后按 Ctrl+C 取消 ping 命令。否则，验证 NAT 实例是否在公有子网中（其路由表有通往互联网网关的路由）。

   ```
   PING ietf.org (104.16.45.99) 56(84) bytes of data.
   64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=1 ttl=33 time=7.88 ms
   64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=2 ttl=33 time=8.09 ms
   64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=3 ttl=33 time=7.97 ms
   ...
   ```

4. 从您的 NAT 实例，使用私有 IP 地址连接到您私有子网中的实例。

   ```
   [ec2-user@ip-10-0-4-184]$ ssh ec2-user@private-server-private-ip-address
   ```

5. 从您的私有实例，通过运行 ping 命令来测试您是否可以连接到互联网：

   ```
   [ec2-user@ip-10-0-135-25]$ ping ietf.org
   ```

   要确认您的私有实例可以通过 NAT 实例访问互联网，请验证您是否收到了如下输出，然后按 Ctrl+C 取消 ping 命令。

   ```
   PING ietf.org (104.16.45.99) 56(84) bytes of data.
   64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=1 ttl=33 time=8.76 ms
   ```

```
64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=2 ttl=33 time=8.26 ms
64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=3 ttl=33 time=8.27 ms
...
```

故障排除

如果从私有子网中的服务器发出的 ping 命令失败，请使用以下步骤来进行问题排查：

- 验证您对启用了 ICMP 的网站执行了 Ping 操作。如果未执行，您的服务器将无法接收应答数据包。要对此进行测试，请从您自己计算机上的命令行终端运行相同的 ping 命令。
- 验证 NAT 实例的安全组是否允许来自私有子网的入站 ICMP 流量。如果不允许，则您的 NAT 实例无法从私有实例接收 ping 命令。
- 验证您对 NAT 实例禁用了源/目的地检查。有关更多信息，请参阅 5. 禁用源/目标检查。
- 验证您正确配置了路由表。有关更多信息，请参阅 6. 更新路由表。

步骤 4：清除

如果您不再需要私有子网中的测试服务器，请终止该实例，这样您就不再为此付费。有关更多信息，请参阅《Amazon EC2 用户指南》中的 终止实例。

如果您不再需要 NAT 实例，则可以停止或终止它，这样您就不再为此付费。如果您创建了一个 NAT AMI，则可以在需要时创建新的 NAT 实例。

比较 NAT 网关和 NAT 实例

以下概要列出了 NAT 实例和 NAT 网关的区别。我们建议您使用 NAT 网关，因为它们提供了更好的可用性和带宽，而且管理工作所需的工作量更少。

属性	NAT 网关	NAT 实例
可用性	高度可用。每个可用区中的 NAT 网关都采用冗余实施。在每个可用区中创建一个 NAT 网关可确保架构不依赖于可用区。	使用脚本管理实例之间的故障转移。
带宽	可以纵向扩展到 100 Gbps。	取决于实例类型的带宽。
维护	由 AWS 管理。您不需要进行任何维护。	由您管理，例如您需要对实例安装软件更新或操作系统补丁。

属性	NAT 网关	NAT 实例
性能	软件经过优化以便处理 NAT 流量。	配置来执行 NAT 的通用 AMI。
费用	费用取决于您使用的 NAT 网关的数量、使用时长以及您通过 NAT 网关发送的数据量。	费用取决于您使用的 NAT 实例的数量、使用时长以及实例类型和大小。
类型和大小	整合提供；您不需要选择类型或范围。	根据您的预测工作负载选择适当的实例类型和大小。
公有 IP 地址	创建时选择弹性 IP 地址，与公有 NAT 网关关联。	为 NAT 实例使用弹性 IP 地址或公有 IP 地址。您随时可以通过将新的弹性 IP 地址与实例关联来更改公有 IP 地址。
私有 IP 地址	在您创建网关时自动从子网的 IP 地址范围中选择。	在您启动实例时，从子网的 IP 地址范围内分配特定的私有 IP 地址。
安全组	您不能将安全组与 NAT 网关相关联。您可以将安全组与 NAT 网关后的资源关联，以控制入站和出站流量。	与您的 NAT 实例和 NAT 实例之后的资源关联，以控制入站和出站流量。
网络 ACL	使用网络 ACL 控制进出您的 NAT 网关所在子网的流量。	使用网络 ACL 控制进出您的 NAT 实例所在子网的流量。
流日志	使用流日志捕获流量。	使用流日志捕获流量。
端口转发	不支持。	手动自定义配置以支持端口转发。
堡垒服务器	不支持。	用作堡垒服务器。
流量指标	查看 NAT 网关的 CloudWatch 指标。	查看实例的 CloudWatch 指标。
超时行为	如果连接超时，NAT 网关向 NAT 网关后方的任何资源返回 RST 数据包，尝试继续进行连接 (它不发送 FIN 数据包)。	如果连接超时，NAT 实例向 NAT 实例后方的资源发送 FIN 数据包，以关闭连接。

属性	NAT 网关	NAT 实例
IP 分段	支持转发 UDP 协议的 IP 分段数据包。 不支持 TCP 和 ICMP 协议的分段。将删除这些协议的分段数据包。	支持重组 UDP、TCP 和 ICMP 协议的 IP 分段数据包。

从 NAT 实例迁移到 NAT 网关

如果您已在使用 NAT 实例，我们建议将它替换为 NAT 网关。您可以在您的 NAT 实例所在的同一子网中创建一个 NAT 网关，然后将路由表中指向该 NAT 实例的现有路由替换为指向该 NAT 网关的路由。要为 NAT 网关使用 NAT 实例当前所用的同一个弹性 IP 地址，您必须先解除该弹性 IP 地址与 NAT 实例的关联，然后在创建 NAT 网关时将该地址与该 NAT 网关关联。

如果您将路由从 NAT 实例更改为 NAT 网关，或者，如果解除弹性 IP 地址与 NAT 实例的关联，则所有当前连接都会中断，必须重新建立。请确保您没有任何关键任务（或任何通过 NAT 实例操作的其他任务）正在运行。

将弹性 IP 地址关联到 VPC 中的资源

弹性 IP 地址是专门针对云计算的动态性质设计的静态、公有 IPv4 地址。此功能有助于将弹性 IP 地址与 AWS 账户中任意虚拟私有云（VPC）内的任何实例或网络接口相关联。您可以利用弹性 IP 地址获得许多优势，从而简化基于云的基础设施的管理和弹性。

弹性 IP 地址的一项主要优势是能遮蔽实例故障。如果实例出现意外中断或需要更换，则可将关联的弹性 IP 地址重新映射到 VPC 中的另一个实例。此失效转移过程可确保应用程序和服务维护一致且可靠的公有端点，从而最大限度地减少停机时间，提供卓越的用户体验。

此外，弹性 IP 地址为管理网络资源的方式提供了灵活性。您可以根据需要以编程方式关联或取消关联这些地址，从而根据不断变化的业务需求将流量引导到不同的实例。这种公有 IP 地址的动态分配让您能够适应不断变化的需求、扩展基础设施以及实现创新的架构，免受静态 IP 分配的限制。

除了用于实例失效转移，弹性 IP 地址还可用作基于云的资源的稳定标识符。在配置外部服务（例如 DNS 记录或防火墙规则）以与 AWS 托管的应用程序通信时，这可能很有用。通过关联永久公有 IP 地址，您可以让自己的联网配置适应未来需求，免于在底层实例被替换或扩展时更新外部引用。

内容

- 弹性 IP 地址概念和规则

- 开始使用弹性 IP 地址

弹性 IP 地址概念和规则

要使用弹性 IP 地址，您必须先将其分配为在您的账户中使用。然后，您可以将其与 VPC 中的实例或网络接口关联。弹性 IP 地址将保持分配到您的AWS账户的状态，直至您明确释放它。

弹性 IP 地址是网络接口的一个属性。您可以通过更新附加到实例的网络接口，将弹性 IP 地址与该实例关联起来。将弹性 IP 地址与网络接口关联而不是直接与实例关联的优势在于，只需一步，即可将网络接口的所有属性从一个实例移至另一个。有关更多信息，请参阅《Amazon EC2 用户指南》中的弹性网络接口。

以下规则适用：

- 弹性 IP 地址一次仅可与一个实例或一个网络接口关联。
- 您可以将弹性 IP 地址从一个实例或网络接口迁移到另一个实例或网络接口。
- 如果将弹性 IP 地址与实例的主要网络接口关联，则系统会将其当前公有 IPv4 地址（如果有）释放到公有 IP 地址池。如果取消关联弹性 IP 地址，则系统将自动在几分钟内为主要网络接口分配一个新的公有 IPv4 地址。如果再向您的实例附加一个网络接口，则不适用此情况。
- 您最多只能拥有 5 个弹性 IP 地址。为帮助节省这些地址，您可以使用 NAT 设备。有关更多信息，请参阅 使用 NAT 设备连接到互联网或其他网络。
- 不支持 IPv6 的弹性 IP 地址。
- 您可以为已分配用于 VPC 的弹性 IP 地址添加标签。不过，不支持成本分配标签。如果您恢复弹性 IP 地址，标签不会恢复。
- 当安全组和网络 ACL 允许来自源 IP 地址的流量时，您可以从 Internet 访问弹性 IP 地址。从 VPC 内部返回到 Internet 的回复流量需要互联网网关。有关更多信息，请参阅安全组 和网络 ACL。
- 您可以对弹性 IP 地址使用以下任一选项：
  - 让 Amazon 提供弹性 IP 地址。选择此选项后，您可以将弹性 IP 地址与网络边界组关联。这是我们通告 CIDR 块的位置。设置网络边界组会将 CIDR 块限制到此组。
  - 使用自带 IP 地址。有关自带 IP 地址的信息，请参阅《Amazon EC2 用户指南》中的自带 IP 地址（BYOIP）。
- 公有 IPv4 地址支持成本分配标签。如果您对弹性 IP 地址应用标签，则可以使用这些标签来跟踪 AWS Cost Explorer 中的公有 IPv4 地址成本。

必须先激活标签，才能将标签用作成本分配标签。有关更多信息，请参阅《AWS Billing 用户指南》中的激活用户定义的成本分配标签。请注意，在您创建用户定义的标签并将其应用到资源之后，标签键最多可能需要 24 小时才能显示在成本分配标签页面上进行激活。

成本分配标签激活后……

- 对于与弹性网络接口关联的所有公有 IPv4 地址（包括分配给 EC2 实例的公有 IPv4 地址和弹性 IP 地址），您可以通过选择使用类型 > PublicIPv4InUseAddress（小时）在 Cost Explorer 中查看与公有 IPv4 地址相关的成本。
- 如果带标签的弹性 IP 地址未与 ENI 关联，或者与已停止的资源（如已停止的 EC2 实例）相关联，则该地址将被视为空闲的 IPv4 地址。您可以通过选择使用类型 > PublicIPv4IdleAddress（小时）在 Cost Explorer 中查看与空闲 IPv4 地址相关的成本。

有关 Cost Explorer 的更多信息，请参阅《AWS Billing 用户指南》中的使用 AWS Cost Explorer 分析费用。

弹性 IP 地址是区域性的。有关使用 Global Accelerator 预置全局 IP 地址的更多信息，请参阅 AWS Global Accelerator 开发人员指南中的使用全局静态 IP 地址而不是区域性静态 IP 地址。

有关弹性 IP 地址定价的更多信息，请参阅 Amazon VPC 定价中的公有 IPv4 地址。

开始使用弹性 IP 地址

以下各节旨在介绍如何开始使用弹性 IP 地址。

任务

- 1. 分配弹性 IP 地址
- 2. 关联弹性 IP 地址
- 3. 解除弹性 IP 地址的关联
- 4. 转移弹性 IP 地址
- 5. 释放弹性 IP 地址
- 6. 恢复弹性 IP 地址
- 命令行概述

1. 分配弹性 IP 地址

要使用弹性 IP 地址，您必须分配一个弹性 IP 地址以在 VPC 中使用。

分配弹性 IP 地址

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Elastic IPs。
3. 选择 Allocate Elastic IP address (分配弹性 IP 地址)。
4. （可选）分配弹性 IP 地址（EIP）时，您可以选择要在其中分配 EIP 的网络边界组。网络边界组是一组可用区（AZ）、Local Zones 或 Wavelength Zones（AWS 可从中公告公有 IP 地址）。Local Zones 和 Wavelength Zones 的网络边界组可能与区域中的可用区不同，以确保 AWS 网络与访问这些区域中资源的客户之间保持最低延迟或最短物理距离。

   > **Important**
   >
   > 您必须在将与该 EIP 关联的 AWS 资源所在的网络边界组中分配一个 EIP。一个网络边界组中的 EIP 只能在该网络边界组中的区域内公告，而不能在由其他网络边界组代表的任何其他区域内公告。

   如果您启用 Local Zones 或 Wavelength Zones（有关更多信息，请参阅启用 Local Zone 或启用 Wavelength Zones），则可以为可用区、Local Zones 或 Wavelength Zones 选择网络边界组。请谨慎选择网络边界组，因为 EIP 及其关联的 AWS 资源必须位于同一个网络边界组中。您可以使用 EC2 控制台查看可用区、Local Zones 或 Wavelength Zones 所在的网络边界组（参阅 Local Zones）。通常，一个区域中的所有可用区都属于同一个网络边界组，而 Local Zones 或 Wavelength Zones 则属于各自独立的网络边界组。

   如果您未启用 Local Zones 或 Wavelength Zones，则在分配 EIP 时，系统会为您预定义代表该区域所有可用区（例如 `us-west-2`）的网络边界组，并且您无法对其进行更改。这意味着您分配给该网络边界组的 EIP 将在您所属区域的所有可用区中公告。
5. 对于公有 IPv4 地址池，选择以下选项之一：
   - Amazon 的 IP 地址池 – 如果要从 Amazon 的 IP 地址池中分配 IPv4 地址。
   - 我的公有 IPv4 地址池 – 如果您想从自己添加到AWS账户的 IP 地址池中分配 IPv4 地址。如果您没有任何 IP 地址池，则此选项将被禁用。
   - 客户拥有的 IPv4 地址池 – 如果要从在本地网络创建的池中分配 IPv4 地址供 Outpost 使用。此选项仅在您拥有 Outpost 时可用。
6. （可选）添加或删除标签。

   [添加标签] 选择添加新标签，然后执行以下操作：

- 对于 Key（键），输入键名称。
- 对于值，输入键值。

[删除标签] 选择标签的“键”和“值”右侧的删除。

7. 选择 Allocate。

2. 关联弹性 IP 地址

您可以将弹性 IP 与 VPC 中正在运行的实例或网络接口相关联。

如果启用了 DNS 主机名，则在您将弹性 IP 地址与实例关联后，实例将会获得公有 DNS 主机名。有关更多信息，请参阅 VPC 中的 DNS 属性。

将弹性 IP 地址与实例或网络接口相关联

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Elastic IPs。
3. 选择已分配用于 VPC（范围列的值为 vpc）的弹性 IP 地址，然后依次选择操作和关联弹性 IP 地址。
4. 选择 Instance 或 Network interface，然后选择实例 ID 或网络接口 ID。选择要与弹性 IP 地址关联的私有 IP 地址。选择 Associate。

3. 解除弹性 IP 地址的关联

要更改弹性 IP 地址所关联的资源，必须先将其与当前关联的资源取消关联。

撤销弹性 IP 地址的关联

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Elastic IPs。
3. 选择该弹性 IP 地址，然后依次选择操作、取消关联弹性 IP 地址。
4. 系统提示时，选择取消关联。

4. 转移弹性 IP 地址

本节介绍如何将弹性 IP 地址从一个 AWS 账户 转移到另一个账户。在以下情况下，转移弹性 IP 地址可能会很有用：

- 组织调整 – 使用弹性 IP 地址转移在 AWS 账户 之间快速转移工作负载。您不必等待新弹性 IP 地址被列入安全组和 NACL 的允许名单。
- 集中式安全管理 – 使用集中式 AWS 安全账户跟踪和转移已通过安全合规性审查的弹性 IP 地址。
- 灾难恢复 – 在紧急事件期间使用弹性 IP 地址转移，为面向公众的互联网工作负载快速重新映射 IP。

转移弹性 IP 地址不收取任何费用。

任务

- 启用弹性 IP 地址转移
- 禁用弹性 IP 地址转移
- 接受转移的弹性 IP 地址

启用弹性 IP 地址转移

本节介绍如何接受转移的弹性 IP 地址。请注意以下与启用弹性 IP 地址转移相关的限制：

- 您可以将弹性 IP 地址从任何 AWS 账户（源账户）转移到同一 AWS 区域的任何其他 AWS 账户（转移账户）。
- 在转移弹性 IP 地址时，AWS 账户 账户之间有两步握手。当源账户开始转移时，转移账户有七天的时间来接受这个弹性 IP 地址转移。在那七天之内，从源账户可以看得见有待转移的那个弹性 IP 地址，比如：在 AWS 控制台中，或通过使用 describe-address-transfers 这个 AWS CLI 命令即可看到。七天之后，转移将过期，而且弹性 IP 地址的所有权将被返回给其原始账户。
- 在接受转移后的 14 天内，源账户可以查看已接受的转移（例如，在 AWS 控制台中或使用 describe-address-transfers AWS CLI 命令查看）。
- AWS 不会向转移账户通知待处理的弹性 IP 地址转移请求。源账户的所有者必须通知转移账户的所有者：他们必须接受弹性 IP 地址转移请求。
- 转移完成后，与正在转移的弹性 IP 地址关联的所有标签都将被重置。
- 您无法将从自带的公有 IPv4 地址池 [通常称为自带 IP（BYOIP）地址池] 分配的弹性 IP 地址转移到您的 AWS 账户。

- 如果您尝试转移与反向 DNS 记录相关联的弹性 IP 地址，则可以开始转移过程，但是在删除关联的 DNS 记录之前，转移账户将无法接受转移。
- 如果已启用并配置了 AWS Outposts，则您可能已从客户拥有的 IP 地址池（CoIP）中分配了弹性 IP 地址。您无法转移从 CoIP 分配的弹性 IP 地址。但是，您可以使用 AWS RAM 与其他账户共享 CoIP。有关更多信息，请参阅《AWS Outposts 用户指南》中的客户拥有的 IP 地址。
- 您可以使用 Amazon VPC IPAM 跟踪向 AWS Organizations 组织中的账户转移弹性 IP 地址的情况。有关更多信息，请参阅查看 IP 地址历史记录。但是，如果将弹性 IP 地址转移到组织外部的 AWS 账户，则弹性 IP 地址的 IPAM 审核历史记录将丢失。

这些步骤必须由源账户完成。

启用弹性 IP 地址转移

1. 确保您使用的是源 AWS 账户。
2. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
3. 在导航窗格中，选择 Elastic IPs。
4. 选择一个或多个要启用转移的弹性 IP 地址，然后选择 Actions（操作）、Enable transfer（启用转移）。
5. 如果要转移多个弹性 IP 地址，则会看到 Transfer type（转移类型）选项。请选择以下选项之一：
   - 如果要将弹性 IP 地址转移到单个 AWS 账户，请选择 Single account（单个账户）。
   - 如果要将弹性 IP 地址转移到多个 AWS 账户，请选择 Multiple accounts（多个账户）。
6. 在 Transfer account ID（转移账户 ID）下，输入要向其转移弹性 IP 地址的 AWS 账户的 ID。
7. 在文本框中输入 **`enable`** 以确认转移。
8. 选择提交。
9. 若要接受转移，请参阅 接受转移的弹性 IP 地址。要禁用转移，请参阅 禁用弹性 IP 地址转移。

禁用弹性 IP 地址转移

本节介绍如何在启用转移后禁用弹性 IP 转移。

这些步骤必须由启用转移的源账户完成。

禁用弹性 IP 地址转移

1. 确保您使用的是源 AWS 账户。

2. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
3. 在导航窗格中，选择 Elastic IPs。
4. 在弹性 IP 的资源列表中，确保启用了显示 Transfer status（转移状态）列的属性。
5. 选择一个或多个 Transfer status（转移状态）为 Pending（待处理）的弹性 IP 地址，然后选择 Actions（操作）、Disable transfer（禁用转移）。
6. 在文本框中输入 **`disable`** 以确认。
7. 选择提交。

接受转移的弹性 IP 地址

本节介绍如何接受转移的弹性 IP 地址。

在转移弹性 IP 地址时，AWS 账户 账户之间有两步握手。当源账户开始转移时，转移账户有七天的时间来接受这个弹性 IP 地址转移。在那七天之内，从源账户可以看得见有待转移的那个弹性 IP 地址，比如：在 AWS 控制台中，或通过使用 describe-address-transfers 这个 AWS CLI 命令即可看到。七天之后，转移将过期，而且弹性 IP 地址的所有权将被返回给其原始账户。

接受转移时，请注意可能发生的以下异常情况以及如何解决这些异常：

- AddressLimitExceeded：如果您的转移账户相应的弹性 IP 地址配额已超出，可以通过源账户来启动弹性 IP 地址转移，但此举将会导致系统出现异常，因为只要是转移账户在尝试接受这样的转移，此异常情况就会发生。默认情况下，每个 AWS 账户的限额为每个区域仅能使用最多 5 个弹性 IP 地址。有关增加限制的说明，请参阅《Amazon EC2 用户指南》中的弹性 IP 地址限制。
- InvalidTransfer.AddressCustomPtrSet：如果您或组织中的某人已将您正在尝试转移的弹性 IP 地址配置为使用反向 DNS 查找，则源账户可以启用弹性 IP 地址转移，但是当转移账户尝试接受转移时，会发生此异常。若要解决此问题，源账户必须删除弹性 IP 地址的 DNS 记录。有关更多信息，请参阅《Amazon EC2 用户指南》中的删除反向 DNS 记录。
- InvalidTransfer.AddressAssociated：如果弹性 IP 地址与 ENI 或 EC2 实例相关联，则源帐户可以启用弹性 IP 地址转移，但是当转移帐户尝试接受转移时，会发生此异常。若要解决此问题，源帐户必须解除弹性 IP 地址的关联。有关更多信息，请参阅《Amazon EC2 用户指南》中的解除弹性 IP 地址的关联。

对于任何其他异常，请联系 支持。

这些步骤必须由转移账户完成。

接受弹性 IP 地址转移

1. 确保您使用的是转移账户。
2. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
3. 在导航窗格中，选择 Elastic IPs。
4. 选择 Actions（操作）、Accept transfer（接受转移）。
5. 当您接受转移时，与要转移的弹性 IP 地址关联的标签不会随弹性 IP 地址一起转移。如果要为所接受的弹性 IP 地址定义 Name（名称）标签，请选择 Create a tag with a key of 'Name' and a value that you specify（创建具有“Name”键以及您指定的值的标签）。
6. 输入要转移的弹性 IP 地址。
7. 如果您接受多个转移的弹性 IP 地址，请选择 Add address（添加地址）以输入附加弹性 IP 地址。
8. 选择提交。

5. 释放弹性 IP 地址

如果您不再需要某个弹性 IP 地址，建议您释放该地址。对于被分配用于 VPC 但未与实例关联的任何弹性 IP 地址，您也需要承担相应费用。弹性 IP 地址不得与实例或网络接口关联。

释放弹性 IP 地址

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Elastic IPs。
3. 选择该弹性 IP 地址，然后依次选择操作、释放弹性 IP 地址。
4. 系统提示时，选择 Release。

6. 恢复弹性 IP 地址

如果您释放了一个弹性 IP 地址，但改变了主意，则可能能够恢复它。如果该弹性 IP 地址已分配给另一个 AWS 账户，则无法恢复该地址，否则恢复它会导致您超出弹性 IP 地址限额。

您可以使用 Amazon EC2 API 或命令行工具恢复弹性 IP 地址。

使用 AWS CLI 恢复弹性 IP 地址

使用 allocate-address 命令，并使用 `--address` 参数指定 IP 地址。

```
aws ec2 allocate-address --domain vpc --address 203.0.113.3
```

命令行概述

您可以使用命令行或 API 执行此部分所述的任务。有关命令行界面的更多信息以及可用 API 操作的列表，请参阅 使用 Amazon VPC。

接受弹性 IP 地址转移

- accept-address-transfer (AWS CLI)
- Approve-EC2AddressTransfer (AWS Tools for Windows PowerShell)

分配弹性 IP 地址

- allocate-address (AWS CLI)
- New-EC2Address (AWS Tools for Windows PowerShell)

将弹性 IP 地址与实例或网络接口关联起来

- associate-address (AWS CLI)
- Register-EC2Address (AWS Tools for Windows PowerShell)

描述弹性 IP 地址转移

- describe-address-transfers (AWS CLI)
- Get-EC2AddressTransfer (AWS Tools for Windows PowerShell)

禁用弹性 IP 地址转移

- disable-address-transfer (AWS CLI)
- Disable-EC2AddressTransfer (AWS Tools for Windows PowerShell)

解除弹性 IP 地址的关联

- disassociate-address (AWS CLI)
- Unregister-EC2Address (AWS Tools for Windows PowerShell)

启用弹性 IP 地址转移

- enable-address-transfer (AWS CLI)
- Enable-EC2AddressTransfer (AWS Tools for Windows PowerShell)

释放弹性 IP 地址

- release-address (AWS CLI)
- Remove-EC2Address (AWS Tools for Windows PowerShell)

为弹性 IP 地址添加标签

- create-tags（AWS CLI）
- New-EC2Tag（AWS Tools for Windows PowerShell）

查看您的弹性 IP 地址

- describe-addresses (AWS CLI)
- Get-EC2Address (AWS Tools for Windows PowerShell)

使用中转网关将您的 VPC 连接到其他 VPC 和网络

您可以使用中转网关来连接 Virtual Private Cloud（VPC）和本地部署的网络，该中转网关将作为中央枢纽，在 VPC、VPN 连接和 AWS Direct Connect 连接之间路由流量。

使用中转网关的一项主要优势是能够集中并简化 VPC 与本地网络之间的连接管理。与其配置多个 VPN 连接或 Direct Connect 链路，不如将中转网关用作单一集成点，因为这有助于降低网络架构的整体复杂性和运营开销。

使用中转网关的定价取决于通过该网关传输的数据量。传入和传出中转网关的数据按 GB 计费，中转网关资源本身则适用单独的按小时计费。具体定价可能因 AWS 区域而异，并且可能会发生变化，因此请务必参考当前 AWS Transit Gateway 定价页面，从中了解最新信息。通过了解中转网关的定价模式，您可以更好地规划和预算与该 AWS 联网服务相关的持续成本。该定价模式加上运营效率和连接优势，使得对寻求构建可扩展且具有成本效益的混合云解决方案的组织而言，中转网关成为极具吸引力的选择。

下表描述了传输网关的一些常见使用场景。有关每个使用场景的更多信息，请参阅《AWS Transit Gateway 用户指南》中的 Example transit gateway scenarios。

示例	用量
集中式路由器	将 Transit Gateway 配置为集中式路由器，以连接所有 VPC、AWS Direct Connect 和 AWS Site-to-Site VPN 连接。
隔离的 VPC	将 Transit Gateway 配置为多个隔离的路由器。这类似于使用多个中转网关，但在路由和连接可能更改的情况下可提供更大的灵活性。
具有共享服务的隔离 VPC	将 Transit Gateway 配置为多个使用共享服务的隔离路由器。这类似于使用多个中转网关，但在路由和连接可能更改的情况下可提供更大的灵活性。

有关更多信息，请参阅 AWS Transit Gateway。

使用 AWS Virtual Private Network 将 VPC 连接到远程网络

您可以使用以下 VPN 选项，来将您的 VPC 连接到远程网络和用户。

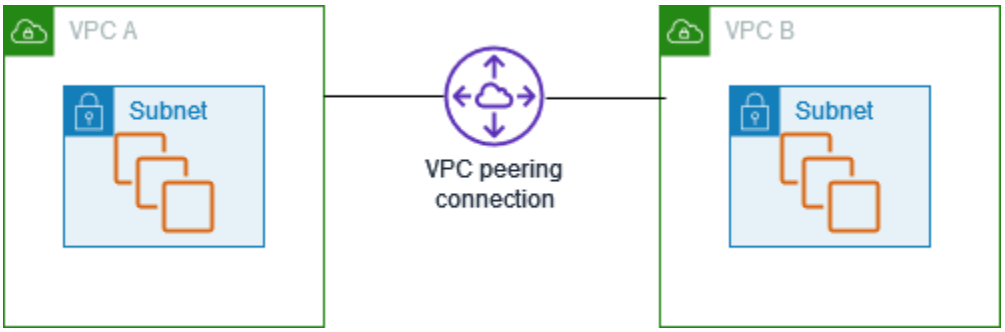
VPN 连接选项	描述
AWS Site-to-Site VPN	您可以在 VPC 和远程网络之间创建 IPsec VPN 连接。在 AWS Site-to-Site VPN 连接的一端，虚拟私有网关或中转网关提供两个 VPN 终端节点（隧道）来进行自动故障转移。您在 Site-to-Site VPN 连接的远程端配置客户网关设备。有关更多信息，请参阅 AWS Site-to-Site VPN 用户指南。
AWS Client VPN	AWS Client VPN 是一种基于客户端的托管 VPN 服务，让您能够安全地访问 AWS 资源或本地部署网络中的资源。借助 AWS Client VPN，您可以配置一个用户可以连接的终端节点，以建立安全的 TLS VPN 会话。这使客户端能够使用基于 OpenVPN 的 VPN 客户端从任何位置访问 AWS 或本地部署中的资源。有关更多信息，请参阅 AWS Client VPN 管理员指南。

VPN 连接选项	描述
AWS VPN CloudHub	如果您拥有多个远程网络（例如，多个分公司），则可以通过虚拟专用网关创建多个 AWS Site-to-Site VPN 连接来启用这些网络之间的通信。有关更多信息，请参阅《AWS Site-to-Site VPN 用户指南》中的使用 VPN CloudHub 在站点之间提供安全通信。
第三方软件 VPN 设备	您可以通过在 VPC 中使用正在运行第三方软件 VPN 应用程序的 Amazon EC2 实例来创建与远程网络的 VPN 连接。AWS 不提供或维护第三方软件 VPN 应用程序；但是，您可以选择合作伙伴和开源社群提供的一系列产品。在 AWS Marketplace 上查找第三方软件 VPN 应用程序。

您还可以使用 AWS Direct Connect 创建远程网络与 VPC 之间的专用私有连接。您可以将此连接与 AWS Site-to-Site VPN 结合来创建经 IPSec 加密的连接。有关更多信息，请参阅 AWS Direct Connect 用户指南 中的什么是 AWS Direct Connect？。

使用 VPC 对等连接来连接 VPC

VPC 对等连接是一项联网功能，支持在 AWS 基础设施内的两个虚拟私有云（VPC）之间进行安全、直接的通信。这种私有连接使资源能够在对等 VPC 中相互交互，如同资源属于同一个网络一样，而无需遍历公共互联网。



创建 VPC 对等连接的过程利用现有 VPC 基础设施来建立此连接，无需网关、AWS Site-to-Site VPN 或任何其他物理硬件。此设计可确保不会出现单点故障或带宽瓶颈。

VPC 对等连接的一项主要优势是能够在不同 AWS 账户之间，甚至在不同 AWS 区域之间连接 VPC。这种灵活性让组织能够无缝集成自有云资源，无论这些资源是在同一个账户内还是分布在多个账户和地理位置。连接的这种私有性质还可确保对等 VPC 之间的所有数据流量都保持在 AWS 网络内，而无需遍历公共互联网。

VPC 对等连接用例的适用范围广泛。组织可以利用此功能在应用程序的不同层（例如 Web 服务器和数据库服务器）之间实现安全通信、促进多个团队或业务部门之间的资源共享，甚至通过将本地网络连接到其 AWS VPC 来实现混合云架构。

VPC 对等连接是两个 VPC 之间的网络连接，您可通过此连接不公开地在这两个 VPC 之间路由流量。对等 VPC 中的资源可以相互通信，如同位于同一网络中。您可以在自己的 VPC 之间，与其他 AWS 账户中的 VPC 或其他 AWS 区域中的 VPC 之间创建 VPC 对等连接。对等 VPC 之间的流量永远不会通过公共互联网传输。

有关更多信息，请参阅 Amazon VPC 对等连接指南。

监控 VPC

您可以使用以下工具来监控 Virtual Private Cloud（VPC）中的流量或网络访问。

Amazon VPC 流日志

您可以使用 VPC 流日志捕获有关您的 VPC 中的网络接口传入和传出流量的详细信息。

Amazon CloudWatch 网络监测仪

您可以使用网络监测仪了解互联网问题如何影响 AWS 上托管的应用程序与最终用户之间的互联网性能和可用性。您还可以近乎实时地探索如何通过切换使用其他服务或通过不同的 AWS 区域 将流量重新路由到工作负载来改善应用程序的预计延迟影响。有关更多信息，请参阅 使用 Amazon CloudWatch 网络监测仪。

Amazon VPC IP 地址管理器（IPAM）

您可以使用 IPAM 来计划、跟踪和监控工作负载的 IP 地址。有关更多信息，请参阅 IP 地址管理器。

流量镜像

您可以使用此功能复制来自 Amazon EC2 实例的弹性网络接口的网络流量，然后将其发送到带外安全和监控设备进行深度数据包检查。您可以检测网络和安全异常情况、获取运营洞察、实施合规性和安全性控制以及排查问题。有关更多信息，请参阅流量镜像。

Reachability Analyzer

您可以使用此工具来分析和调试 VPC 中两个资源之间的网络连接。指定源资源和目标资源后，如果可以访问这些资源，则 Reachability Analyzer 会生成有关这些资源间的虚拟路径的逐跳详细信息，如果无法访问，则会确定导致无法连接的障碍组件。有关更多信息，请参阅 Reachability Analyzer。

网络访问分析器

您可以使用网络访问分析器来了解对资源的网络访问。这可帮助您确定在网络安保状况方面需要的改进，以及证明您的网络符合特定的合规性要求。有关更多信息，请参阅网络访问分析器。

CloudTrail 日志

AWS CloudTrail 记录 Amazon VPC 的 API 调用，例如：

- 进行了哪些 API 调用（比如创建或修改 VPC 资源之类的操作）
- 调用的源 IP 地址
- 调用方

- 调用时间

为 CreateVpc、DeleteVpc 和 CreateDefaultVpc 操作创建了单独的日志。这些日志还包括所创建的 VPC 相关默认资源（如任何默认互联网网关或默认安全组）。

有关更多信息，请参阅《Amazon EC2 用户指南》中的使用 AWS CloudTrail 记录 Amazon EC2 API 调用。

使用 VPC 流日志记录 IP 流量

利用 VPC 流日志这项功能，您可以捕获有关传入和传出您的 VPC 中网络接口的 IP 流量的信息。流日志数据可发布到以下位置：Amazon CloudWatch Logs、Amazon S3 或 Amazon Data Firehose。允许将网络流量日志发送到 CloudWatch Logs 或 S3 等目标的已配置传输路径和权限称为订阅。创建流日志后，您可以在配置的日志组、存储桶或传输流中检索和查看流日志记录。

流日志可帮助您处理多种任务，例如：

- 诊断过于严格的安全组规则
- 监控达到您实例的流量
- 确定在网络接口上往返的流量的方向

流日志数据的收集在您网络流量路径之外，因此不会影响网络吞吐量或延迟。您可以创建或删除流日志，而不会对网络性能造成任何影响。

> **Note**
>
> 本节仅讨论 VPC 的流日志。有关版本 6 中引入的中转网关流日志的信息，请参阅《Amazon VPC 中转网关用户指南》中的使用中转网关流日志记录网络流量。

内容

- 流日志基础知识
- 流日志记录
- 流日志记录示例
- 流日志限制
- 定价

- 使用流日志
- 将流日志发布到 CloudWatch Logs
- 将流日志发布到 Amazon S3
- 将流日志发布到 Amazon Data Firehose
- 使用 Amazon Athena 查询流日志
- VPC 流日志疑难解答

流日志基础知识

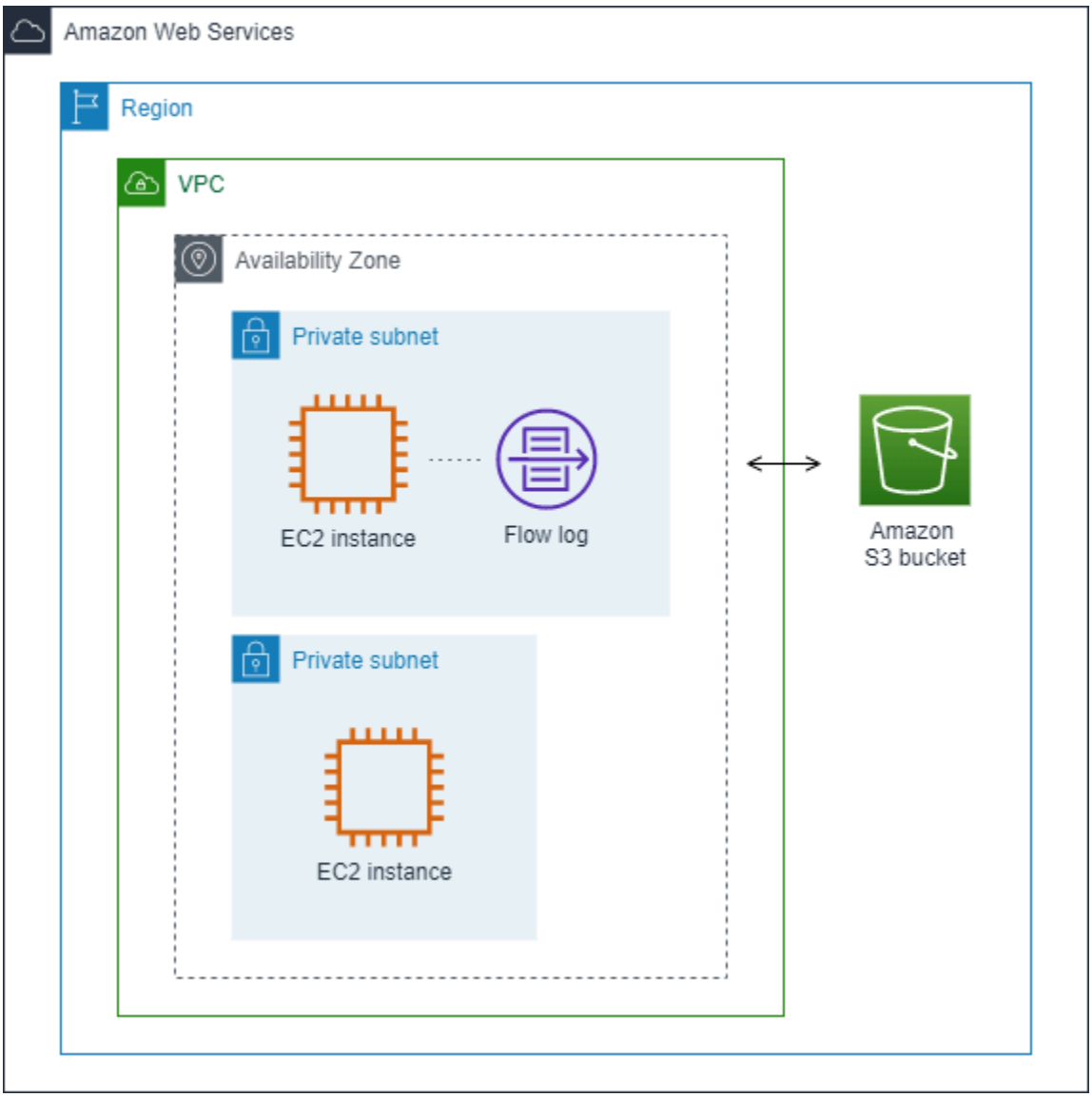
您可以为 VPC、子网或网络接口创建流日志。如果您为子网或 VPC 创建流日志，则会监视该子网或 VPC 中的每个网络接口。

受监控网络接口的流日志数据记录为流日志记录，这些是日志事件，由描述该流量的字段组成。有关更多信息，请参阅 流日志记录。

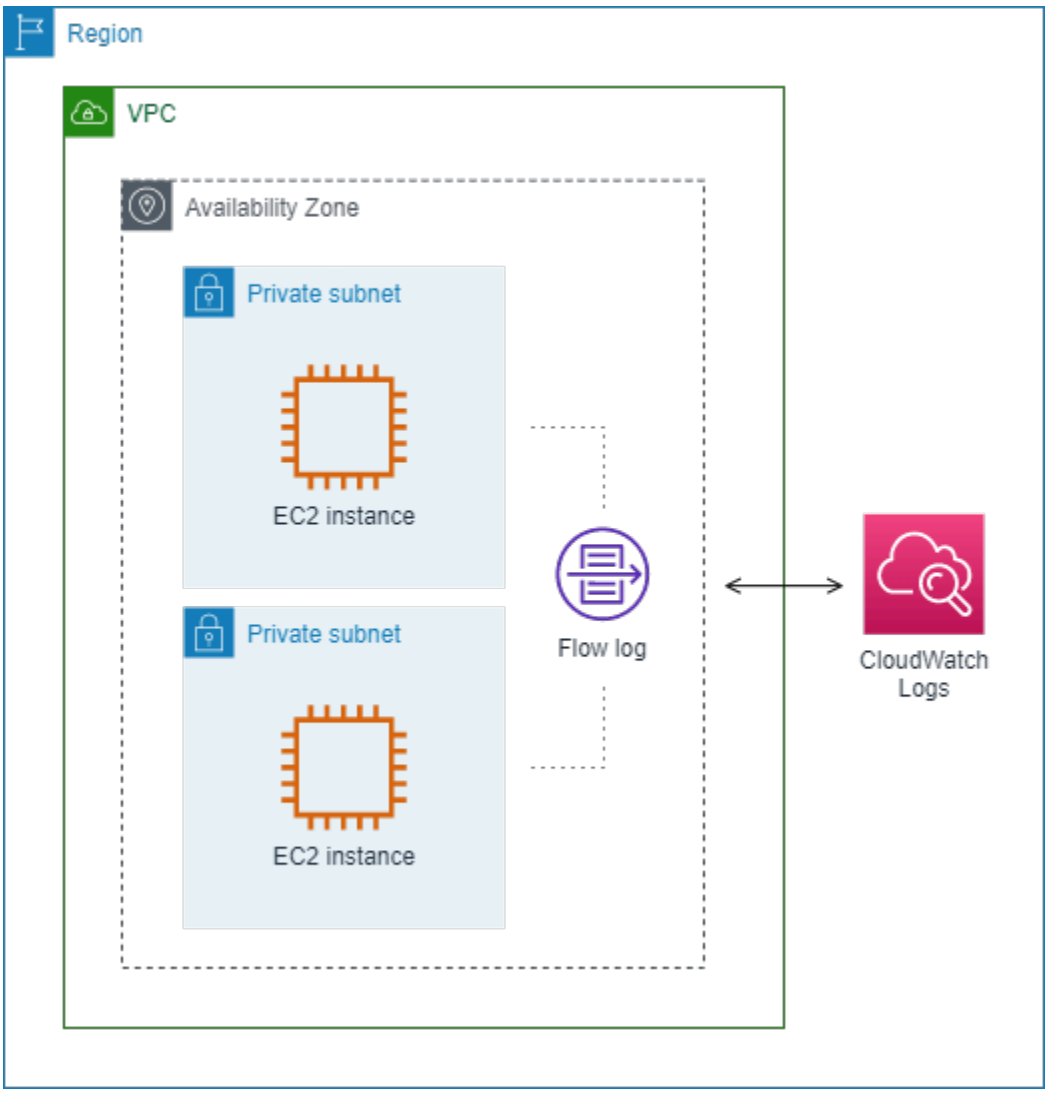
要创建流日志，请指定：

- 要为其创建流日志的资源
- 要捕获的流量的类型（接受的流量、拒绝的流量或所有流量）
- 指定您要将流日志数据发布到的目标

在以下示例中，您创建一个流日志，用于捕获私有子网中某个 EC2 实例的网络接口的已接受流量，并将流日志记录发布到 Amazon S3 存储桶。



在以下示例中，流日志会捕获子网的所有流量，并将流日志记录发布到 Amazon CloudWatch Logs。流日志将捕获子网中所有网络接口的流量。



创建流日志后，需要几分钟来开始收集数据并将数据发布到选定目标。流日志不会为您的网络接口捕获实时日志流。有关更多信息，请参阅 2. 创建流日志。

如果在为子网或 VPC 创建流日志后，您在子网中启动了实例，则只要网络接口中有网络流量，我们就会为新网络接口创建一个日志流（对于 CloudWatch Logs）或日志文件对象（对于 Amazon S3）。

您可以为其他 AWS 创建的网络接口创建流日志，例如：

- Elastic Load Balancing
- Amazon RDS
- Amazon ElastiCache
- Amazon Redshift
- Amazon WorkSpaces

- NAT 网关
- 中转网关

不论网络接口是什么类型，您必须使用 Amazon EC2 控制台或 Amazon EC2 API 为网络接口创建流日志。

您可以将标签应用于流日志。每个标签都包含您定义的一个键和一个可选值。标签可以帮助您整理流日志，例如按目的或拥有者。

如果您不再需要某个流日志，可将其删除。删除流日志会禁用资源的流日志服务，因此将不再创建或发布新的流日志记录。删除流日志不会删除任何现有的流日志数据。删除流日志后，您可以将其从目的地中直接删除。有关更多信息，请参阅 4. 删除流日志。

流日志记录

流日志记录代表您的 VPC 中的网络流。默认情况下，每条记录捕获在聚合时间间隔（又称为捕获窗口）内发生的网络 Internet 协议 (IP) 流量流（按每个网络接口 5 元组来定性）。

每条记录都是一个字符串，字段用空格分隔。记录包括 IP 流的不同组件的值，包括源、目标和协议。

当您创建流日志时，您可以为流日志记录使用默认格式，也可以指定自定义格式。

目录

- 聚合时间间隔
- 默认格式
- 自定义格式
- 可用字段

聚合时间间隔

聚合时间间隔表示捕获特定流并聚合到流日志记录中的时间段。默认情况下，最大聚合时间间隔为 10 分钟。创建流日志时，您可以选择指定最大 1 分钟的聚合时间间隔。最大聚合时间间隔为 1 分钟的流日志的大小，比最大聚合时间间隔为 10 分钟的流日志大。

当网络接口附加到基于 Nitro 的实例时，无论指定的最大聚合时间间隔为多少，聚合时间间隔始终不超过 1 分钟。

在聚合时间间隔内捕获数据后，需要额外的时间来处理数据并将其发布到 CloudWatch Logs 或 Amazon S3。流日志服务通常在大约 5 分钟内将日志传送到 CloudWatch Logs，在大约 10 分钟内将

日志传送到 Amazon S3。但是，日志交付已尽了最大努力，您的日志可能会延迟到典型交付时间之后。

默认格式

使用默认格式，流日志记录按可用字段表中显示的顺序包括版本 2 字段。您无法自定义或更改默认格式。要捕获其他字段或不同字段子集，请指定自定义格式。

自定义格式

使用自定义格式，您可以指定流日志记录中包含哪些字段以及采用哪种顺序。这使您可以根据具体需求创建流日志，并忽略无关的字段。使用自定义格式，还可减少从发布的流日志提取特定信息所需的单独流程。您可以指定任意数量的可用流日志字段，但必须至少指定一个。

可用字段

下表描述了对流日志记录可用的所有字段。版本列表示在其中引入了字段的 VPC 流日志版本。默认格式包括所有版本 2 字段，与它们在表格中出现的顺序相同。

将流日志数据发布到 Amazon S3 时，字段的数据类型将取决于流日志格式。如果格式为纯文本，则所有字段的类型均为 STRING。如果格式为 Parquet，请参阅字段数据类型表。

如果某个字段不适用于或无法计算特定记录，则记录为该条目显示一个“-”符号。不直接来自数据包标头的元数据字段是最大努力的近似值，它们的值可能缺失或不准确。

字段	描述	版本
version	VPC 流日志版本。如果您使用默认格式，则版本为 2。如果您使用自定义格式，则版本是指定字段中的最高版本。例如，如果您只在版本 2 中指定字段，则版本为 2。如果您在版本 2、3 和 4 中指定字段组合，则版本为 4。 Parquet 数据类型：INT_32	2
account-id	为其记录流量的源网络接口的拥有者的AWS账户 ID。如果该网络接口由 AWS 服务创建，例如在创建 VPC 端点或网络负载均衡器时创建，则记录中可能会显示此字段的值为 unknown。 Parquet 数据类型：STRING	2
interface-id	为其记录流量的网络接口的 ID。	2

字段	描述	版本
	Parquet 数据类型：STRING	
srcaddr	对于传入流量，这是流量来源的 IP 地址。对于传出流量，这是发送流量的网络接口的私有 IPv4 地址或 IPv6 地址。另请参阅pkt-srcaddr。 Parquet 数据类型：STRING	2
dstaddr	传出流量的目标地址，或者网络接口上传入流量的网络接口 IPv4 或 IPv6 地址。网络接口的 IPv4 地址始终是其私有 IPv4 地址。另请参阅 pkt-dstaddr。 Parquet 数据类型：STRING	2
srcport	流量的源端口。 Parquet 数据类型：INT_32	2
dstport	流量的目标端口。 Parquet 数据类型：INT_32	2
protocol	流量的 IANA 协议编号。有关更多信息，请参阅分配的 Internet 协议编号。 Parquet 数据类型：INT_32	2
packets	在流中传输的数据包的数量。 Parquet 数据类型：INT_64	2
bytes	在流中传输的字节数。 Parquet 数据类型：INT_64	2
start	在聚合时间间隔内，接收流的第一个数据包的时间（以 Unix 秒为单位）。这可能是在网络接口上传输或收到数据包之后最多 60 秒。 Parquet 数据类型：INT_64	2

字段	描述	版本
end	在聚合时间间隔内，接收流的最后一个数据包的时间（以 Unix 秒为单位）。这可能是在网络接口上传输或收到数据包之后最多 60 秒。 Parquet 数据类型：INT_64	2
action	与流量关联的操作： • ACCEPT — 流量已被接受。 • REJECT — 流量已被拒绝。例如，安全组或网络 ACL 不允许流量，或数据包在连接关闭之后到达。 Parquet 数据类型：STRING	2
log-status	流日志的日志记录状态： • OK – 数据正常记录到选定目标。 • NODATA – 聚合时间间隔内没有传入或传出网络接口的网络流量。 • SKIPDATA – 在聚合时间间隔内跳过了一些流日志记录。这可能是由于内部容量限制或内部错误。 在聚合时间间隔内可能会跳过一些流日志记录（请参阅可用字段中的 log-status）。这可能是因为存在内部 AWS 容量限制或内部错误。如果使用 AWS Cost Explorer 查看 VPC 流日志费用，并且在流日志聚合时间间隔内跳过了一些流日志，则 AWS Cost Explorer 中报告的流日志数量会高于 Amazon VPC 发布的流日志数量。 Parquet 数据类型：STRING	2
vpc-id	包含记录其流量的网络接口的 VPC 的 ID。 Parquet 数据类型：STRING	3
subnet-id	包含记录其流量的网络接口的子网的 ID。 Parquet 数据类型：STRING	3

字段	描述	版本
instance-id	与要记录其流量的网络接口关联的实例的 ID（如果实例由您所有）。对于请求方管理的网络接口，返回“-”符号，例如，NAT 网关的网络接口。 Parquet 数据类型：STRING	3
tcp-flags	以下 TCP 标志的位掩码值： • FIN — 1 • SYN — 2 • RST — 4 • SYN-ACK – 18 如果未记录支持的标志，则 TCP 标志值为 0。例如，由于 tcp-flags 不支持记录 ACK 或 PSH 标志，因此具有这些不受支持标志的流量记录将导致 tcp-flags 值为 0。但是，如果不支持的标志附带有支持标志，我们将报告受支持标志的值。例如，如果 ACK 是 SYN-ACK 的一部分，则会报告 18。而且，如果有像 SYN+ECE 这样的记录，由于 SYN 是支持的标志，而 ECE 不是，那么 TCP 标志值为 2。如果由于某种原因标志组合无效且无法计算其值，则值为“-”。如果未发送标志，则 TCP 标志值为 0。 在聚合时间间隔内，TCP 标志可以是 OR-ed。对于短连接，标志必须在与流日志记录相同的行上设置，例如，对于 SYN-ACK 和 FIN 的 19，以及对于 SYN 和 FIN 的 3。有关示例，请参阅 TCP 标志序列。 有关 TCP 标志的一般信息（例如 FIN、SYN 和 ACK 等标志的含义），请参阅 Wikipedia 上的 TCP 分段结构。 Parquet 数据类型：INT_32	3
type	流量的类型。可能的值包括：IPv4 | IPv6 | EFA。有关更多信息，请参阅 Elastic Fabric Adapter。 Parquet 数据类型：STRING	3

字段	描述	版本
pkt-srcaddr	流量的数据包级别（原始）源 IP 地址。将此字段与 srcaddr 字段一起使用，用于区分流量流经的中间层 IP 地址与流量的原始源 IP 地址。例如，当流量流经 NAT 网关的网络接口时，或者当 Amazon EKS 中 Pod 的 IP 地址不同于运行 Pod 的实例节点的网络接口 IP 地址时（用于 VPC 中的通信）。 Parquet 数据类型：STRING	3
pkt-dstaddr	流量的数据包级别（原始）目标 IP 地址。将此字段与 dstaddr 字段一起使用，用于区分流量流经的中间层的 IP 地址与流量的最终目标 IP 地址。例如，当流量流经 NAT 网关的网络接口时，或者当 Amazon EKS 中 Pod 的 IP 地址不同于运行 Pod 的实例节点的网络接口 IP 地址时（用于 VPC 中的通信）。 Parquet 数据类型：STRING	3
region	包含记录其流量的网络接口的区域。 Parquet 数据类型：STRING	4
az-id	包含记录其流量的网络接口的可用区的 ID。如果流量来自子位置，则记录会对此字段显示“-”符号。 Parquet 数据类型：STRING	4
sublocation-type	sublocation-id 字段中返回的子位置类型：可能的值包括：波长 | 前哨 | 本地扩展区。如果流量不是来自子位置，则记录会对此字段显示“-”符号。 Parquet 数据类型：STRING	4
sublocation-id	包含记录其流量的网络接口的子位置的 ID。如果流量不是来自子位置，则记录会对此字段显示“-”符号。 Parquet 数据类型：STRING	4

字段	描述	版本
pkt-src-aws-service	pkt-srcaddr 字段的 IP 地址范围子集的名称（如果源 IP 地址用于AWS服务）。如果源 IP 地址属于重叠范围，则 pkt-src-aws-service 仅显示其中一个 AWS 服务代码。可能的值包括：AMAZON | AMAZON_APPFLOW | AMAZON_CONNECT | API_GATEWAY | AURORA_DSQL | CHIME_MEETINGS | CHIME_VOICECONNECTOR | CLOUD9 | CLOUDFRONT | CLOUDFRONT_ORIGIN_FACING | CODEBUILD | DYNAMODB | EBS | EC2 | EC2_INSTANCE_CONNECT | GLOBALACCELERATOR | IVS_LOW_LATENCY | IVS_REALTIME | KINESIS_VIDEO_STREAMS | MEDIA_PACKAGE_V2 | ROUTE53 | ROUTE53_HEALTHCHECKS | ROUTE53_HEALTHCHECKS_PUBLISHING | ROUTE53_RESOLVER | S3 | WORKSPACES_GATEWAYS 。 Parquet 数据类型：STRING	5
pkt-dst-aws-service	pkt-dstaddr 字段的 IP 地址范围子集的名称（如果目标 IP 地址用于AWS服务）。有关可能的值的列表，请参阅 pkt-src-aws-service 字段。 Parquet 数据类型：STRING	5
flow-direction	相对于捕获流量的接口而言流的方向。可能的值包括：ingress | egress。 Parquet 数据类型：STRING	5

字段	描述	版本
traffic-path	传出流量到达目的地的路径。要确定流量是否为传出流量，请检查 flow-direction 字段。可能的值如下所示。如果没有任何值适用，则该字段将设置为 -。 • 1 – 通过同一 VPC 中的其他资源，包括在该 VPC 中创建网络接口的资源 • 2 — 通过互联网网关或网关 VPC 端点 • 3 — 通过虚拟私有网关 • 4 — 通过区域内 VPC 对等连接 • 5 — 通过区域间 VPC 对等连接 • 6 — 通过本地网关 • 7 — 通过网关 VPC 端点（仅限基于 Nitroc 的实例） • 8 — 通过互联网网关（仅限基于 Nitro 的实例） Parquet 数据类型：INT_32	5
ecs-cluster-arn	如果流量来自正在运行的 ECS 任务，则为 ECS 集群的 AWS 资源名称（ARN）。要在订阅中包含此字段，您需要调用 ecs:ListClusters 的权限。 Parquet 数据类型：STRING	7
ecs-cluster-name	如果流量来自正在运行的 ECS 任务，则为 ECS 集群的名称。要在订阅中包含此字段，您需要调用 ecs:ListClusters 的权限。 Parquet 数据类型：STRING	7
ecs-container-instance-arn	如果流量来自 EC2 实例上正在运行的 ECS 任务，则为 ECS 容器实例的 ARN。如果容量提供程序是 AWS Fargate，则此字段将为“-”。要在订阅中包含此字段，您需要调用 ecs:ListClusters 和 ecs:ListContainerInstances 的权限。 Parquet 数据类型：STRING	7

字段	描述	版本
ecs-container-instance-id	如果流量来自 EC2 实例上正在运行的 ECS 任务，则为 ECS 容器实例的 ID。如果容量提供程序是 AWS Fargate，则此字段将为“-”。要在订阅中包含此字段，您需要调用 ecs:ListClusters 和 ecs:ListContainerInstances 的权限。 Parquet 数据类型：STRING	7
ecs-container-id	如果流量来自正在运行的 ECS 任务，则为容器的 Docker 运行时 ID。如果 ECS 任务中有一个或多个容器，这将是第一个容器的 docker 运行时 ID。要在订阅中包含此字段，您需要调用 ecs:ListClusters 的权限。 Parquet 数据类型：STRING	7
ecs-second-container-id	如果流量来自正在运行的 ECS 任务，则为容器的 Docker 运行时 ID。如果 ECS 任务中有多个容器，这将是第二个容器的 Docker 运行时 ID。要在订阅中包含此字段，您需要调用 ecs:ListClusters 的权限。 Parquet 数据类型：STRING	7
ecs-service-name	如果流量来自正在运行的 ECS 任务，并且 ECS 任务由 ECS 服务启动，则为 ECS 服务的名称。如果 ECS 任务不是由 ECS 服务启动的，则此字段将为“-”。要在订阅中包含此字段，您需要调用 ecs:ListClusters 和 ecs:ListServices 的权限。 Parquet 数据类型：STRING	7
ecs-task-definition-arn	如果流量来自正在运行的 ECS 任务，则为 ECS 任务定义的 ARN。要在订阅中包含此字段，您需要调用 ecs:ListClusters 和 ecs:ListTaskDefinitions 的权限 Parquet 数据类型：STRING	7
ecs-task-arn	如果流量来自正在运行的 ECS 任务，则为 ECS 任务的 ARN。要在订阅中包含此字段，您需要调用 ecs:ListClusters 和 ecs:ListTasks 的权限。 Parquet 数据类型：STRING	7

字段	描述	版本
ecs-task-id	如果流量来自正在运行的 ECS 任务，则为 ECS 任务的 ID。要在订阅中包含此字段，您需要调用 ecs:ListClusters 和 ecs:ListTasks 的权限。 Parquet 数据类型：STRING	7
reject-reason	流量被拒绝的原因。可能的值：BPA。对于任何其他拒绝原因，返回“-”。有关 VPC 屏蔽公共访问权限（BPA）的更多信息，请参阅 屏蔽 VPC 和子网的公共访问权限。 Parquet 数据类型：STRING	8

流日志记录示例

以下是捕获特定流量流的流日志记录的示例。

有关流日志记录格式的信息，请参阅 流日志记录。有关如何创建流日志的信息，请参阅 使用流日志。

目录

- 接受的和拒绝的流量
- 无数据和跳过的记录
- 安全组和网络 ACL 规则
- IPv6 流量
- TCP 标志序列
- 通过 NAT 网关的流量
- 通过中转网关的流量
- 服务名称、流量路径和流向

接受的和拒绝的流量

以下是默认流日志记录的示例。

在本例中，允许从 IP 地址 172.31.16.139 到具有私有 IP 地址为 172.31.16.21 的网络接口的 SSH 流量（目标端口 22，TCP 协议），并允许账户 123456789010 中的 ID eni-1235b8ca123456789。

```
2 123456789010 eni-1235b8ca123456789 172.31.16.139 172.31.16.21 20641 22 6 20 4249
 1418530010 1418530070 ACCEPT OK
```

在此示例中，拒绝指向账户 123456789010 中的网络接口 eni-1235b8ca123456789 的 RDP 流量（目标端口 3389，TCP 协议）。

```
2 123456789010 eni-1235b8ca123456789 172.31.9.69 172.31.9.12 49761 3389 6 20 4249
 1418530010 1418530070 REJECT OK
```

无数据和跳过的记录

以下是默认流日志记录的示例。

在此示例中，聚合时间间隔中未记录任何数据。

```
2 123456789010 eni-1235b8ca123456789 - - - - - - - 1431280876 1431280934 - NODATA
```

由于超过内部容量，VPC 流日志在聚合时间间隔期间无法捕获流日志数据时会跳过记录。单个跳过的记录可以表示聚合时间间隔期间网络接口未捕获的多个流。

```
2 123456789010 eni-11111111aaaaaaaaa - - - - - - - 1431280876 1431280934 - SKIPDATA
```

> **Note**
>
> 在聚合时间间隔内可能会跳过一些流日志记录（请参阅可用字段中的 log-status）。这可能是因为存在内部 AWS 容量限制或内部错误。如果使用 AWS Cost Explorer 查看 VPC 流日志费用，并且在流日志聚合时间间隔内跳过了一些流日志，则 AWS Cost Explorer 中报告的流日志数量会高于 Amazon VPC 发布的流日志数量。

安全组和网络 ACL 规则

如果您正使用流日志来诊断过于严格或过于宽松的安全组规则或网络 ACL 规则，请注意这些资源的状态性。安全组是有状态的 — 这意味着对所允许流量的响应也会被允许，即使安全组中的规则不允许也是如此。相反，网络 ACL 是无状态的，因此对所允许流量的响应需要遵守网络 ACL 规则。

例如，您从家中的计算机（IP 地址为 203.0.113.12）对您的实例（网络接口的私有 IP 地址为 172.31.16.139）使用 ping 命令。您的安全组入站规则允许 ICMP 流量，但出站规则不允许 ICMP 流

量。由于安全组是有状态的，允许从您的实例响应 ping。您的网络 ACL 允许入站 ICMP 流量，但不允许出站 ICMP 流量。由于网络 ACL 是无状态的，响应 Ping 将被丢弃，不会传输到您家中的计算机。在默认流日志中，它显示为两个流日志记录：

- 网络 ACL 和安全组都允许 (因此可到达您的实例) 的发起 ping 的 ACCEPT 记录。
- 网络 ACL 拒绝的响应 ping 的 REJECT 记录。

```
2 123456789010 eni-1235b8ca123456789 203.0.113.12 172.31.16.139 0 0 1 4 336 1432917027
 1432917142 ACCEPT OK
```

```
2 123456789010 eni-1235b8ca123456789 172.31.16.139 203.0.113.12 0 0 1 4 336 1432917094
 1432917142 REJECT OK
```

如果您的网络 ACL 允许出站 ICMP 流量，流日志会显示两个 ACCEPT 记录 (一个针对发起 ping，一个针对响应 ping)。如果您的安全组拒绝入站 ICMP 流量，流日志会显示一个 REJECT 记录，因为流量无权到达您的实例。

IPv6 流量

以下是默认流日志记录的示例。在此示例中，允许从 IPv6 地址 2001:db8:1234:a100:8d6e:3477:df66:f105 到账户 123456789010 中的网络接口 eni-1235b8ca123456789 的 SSH 流量（端口 22）。

```
2 123456789010 eni-1235b8ca123456789 2001:db8:1234:a100:8d6e:3477:df66:f105
 2001:db8:1234:a102:3304:8879:34cf:4071 34892 22 6 54 8855 1477913708 1477913820 ACCEPT
 OK
```

TCP 标志序列

本节包含按照下列顺序捕获下列字段的自定义流日志的示例。

```
version vpc-id subnet-id instance-id interface-id account-id type srcaddr dstaddr
 srcport dstport pkt-srcaddr pkt-dstaddr protocol bytes packets start end action tcp-
flags log-status
```

本节示例中的 tcp-flags 字段显示为流日志中的倒数第二的值。TCP 标志可帮助您确定流量的方向，例如，启动连接的服务器。

> **Note**
>
> 有关 tcp-flags 选项的更多信息和每个 TCP 标志的说明，请参阅 可用字段。

在以下记录（从晚上 7:47:55 开始，到晚上 7:48:53 结束）中，运行在端口 5001 上的客户端启动了两个与服务器的连接。服务器从客户端上的两个不同端口（43416 和 43418）收到两个 SYN 标志 (2)。对于每个 SYN，在对应的端口上从服务器向客户端发送一个 SYN-ACK (18)。

```
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
 eni-1235b8ca123456789 123456789010 IPv4 52.213.180.42 10.0.0.62 43416 5001
 52.213.180.42 10.0.0.62 6 568 8 1566848875 1566848933 ACCEPT 2 OK
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
 eni-1235b8ca123456789 123456789010 IPv4 10.0.0.62 52.213.180.42 5001 43416 10.0.0.62
 52.213.180.42 6 376 7 1566848875 1566848933 ACCEPT 18 OK
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
 eni-1235b8ca123456789 123456789010 IPv4 52.213.180.42 10.0.0.62 43418 5001
 52.213.180.42 10.0.0.62 6 100701 70 1566848875 1566848933 ACCEPT 2 OK
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
 eni-1235b8ca123456789 123456789010 IPv4 10.0.0.62 52.213.180.42 5001 43418 10.0.0.62
 52.213.180.42 6 632 12 1566848875 1566848933 ACCEPT 18 OK
```

在第二个聚合时间间隔中，上一个流期间建立的连接之一现在关闭。服务器在端口 43418 上为连接发送 FIN 标志 (1) 到客户端。客户端在端口 43418 上发送 FIN 到服务器。

```
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
 eni-1235b8ca123456789 123456789010 IPv4 10.0.0.62 52.213.180.42 5001 43418 10.0.0.62
 52.213.180.42 6 63388 1219 1566848933 1566849113 ACCEPT 1 OK
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
 eni-1235b8ca123456789 123456789010 IPv4 52.213.180.42 10.0.0.62 43418 5001
 52.213.180.42 10.0.0.62 6 23294588 15774 1566848933 1566849113 ACCEPT 1 OK
```

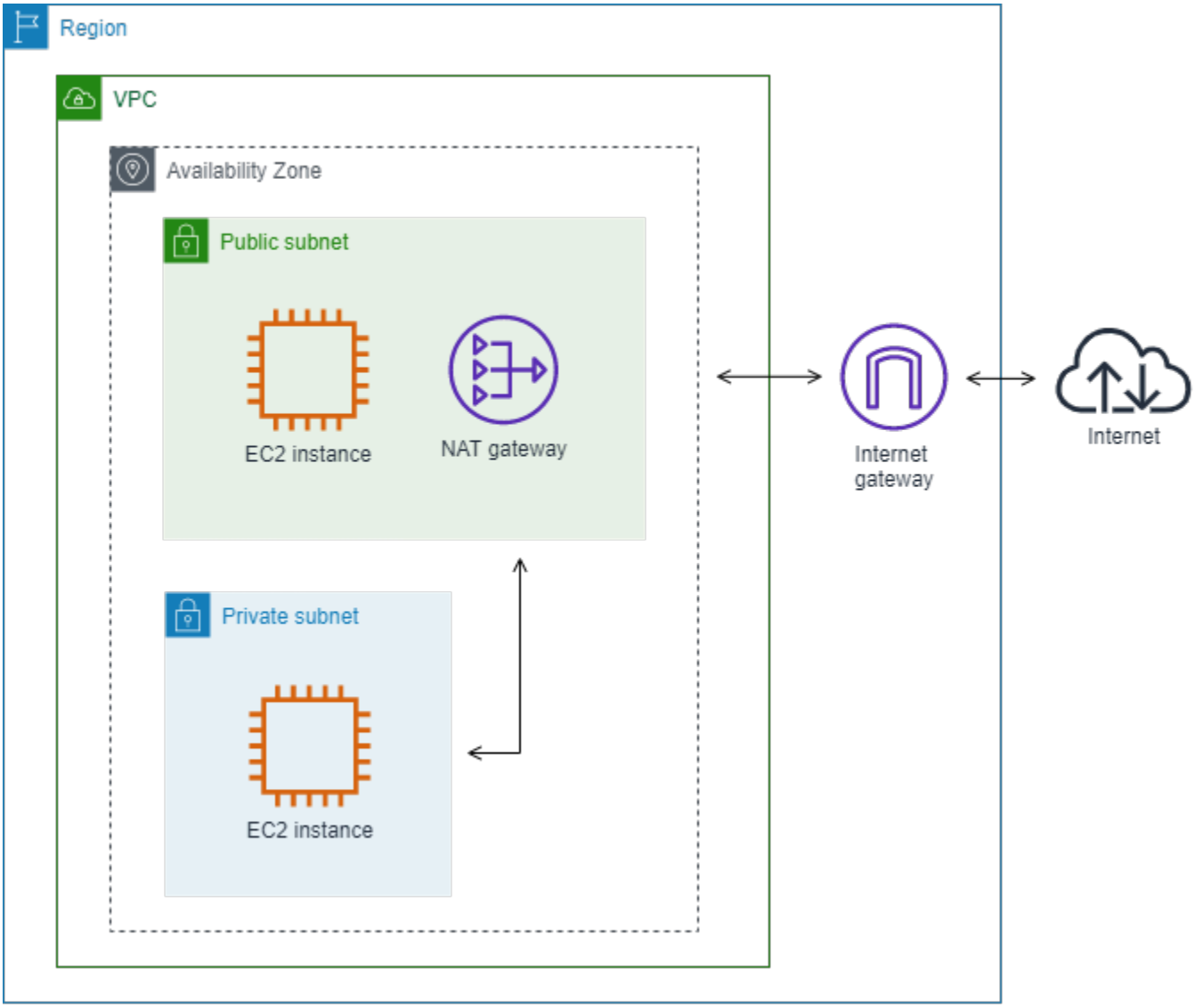
对于在单个聚合时间间隔中打开和关闭的短连接（例如，几秒），对于处于相同方向的流量流，标志必须在与流日志记录相同的行上设置。在以下示例中，连接在相同的聚合时间间隔中建立和完成。在第一行，TCP 标志值为 3，这表示有从客户端发送到服务器的 SYN 和 FIN 消息。在第二行，TCP 标志值为 19，这表示有从服务器发送回客户端的 SYN-ACK 和 FIN 消息。

```
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
 eni-1235b8ca123456789 123456789010 IPv4 52.213.180.42 10.0.0.62 43638 5001
 52.213.180.42 10.0.0.62 6 1260 17 1566933133 1566933193 ACCEPT 3 OK
```

```
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
 eni-1235b8ca123456789 123456789010 IPv4 10.0.0.62 52.213.180.42 5001 43638  10.0.0.62
 52.213.180.42 6 967 14 1566933133 1566933193 ACCEPT 19 OK
```

通过 NAT 网关的流量

在此示例中，私有子网中的实例通过位于公有子网中的 NAT 网关访问 Internet。



NAT 网关网络接口的以下自定义流日志按照下列顺序捕获下列字段。

```
instance-id interface-id srcaddr dstaddr pkt-srcaddr pkt-dstaddr
```

流日志显示流量从实例 IP 地址 (10.0.1.5) 通过 NAT 网关网络接口流向 Internet 上的主机 (203.0.113.5)。NAT 网关网络接口是请求方管理的网络接口，因此流量日志记录为 instance-id 字段显

示“-”符号。以下行显示从源实例流向 NAT 网关网络接口的流量。dstaddr 和 pkt-dstaddr 字段的值不同。dstaddr 字段显示 NAT 网关网络接口的私有 IP 地址，pkt-dstaddr 字段显示 Internet 上主机的最终目标 IP 地址。

```
- eni-1235b8ca123456789 10.0.1.5 10.0.0.220 10.0.1.5 203.0.113.5
```

接下来两行显示从 NAT 网关网络接口流向 Internet 上目标主机的流量，以及从主机发送到 NAT 网关网络接口的响应流量。

```
- eni-1235b8ca123456789 10.0.0.220 203.0.113.5 10.0.0.220 203.0.113.5
- eni-1235b8ca123456789 203.0.113.5 10.0.0.220 203.0.113.5 10.0.0.220
```

以下行显示从 NAT 网关网络接口流向源实例的响应流量。srcaddr 和 pkt-srcaddr 字段的值不同。srcaddr 字段显示 NAT 网关网络接口的私有 IP 地址，pkt-srcaddr 字段显示 Internet 上主机的 IP 地址。

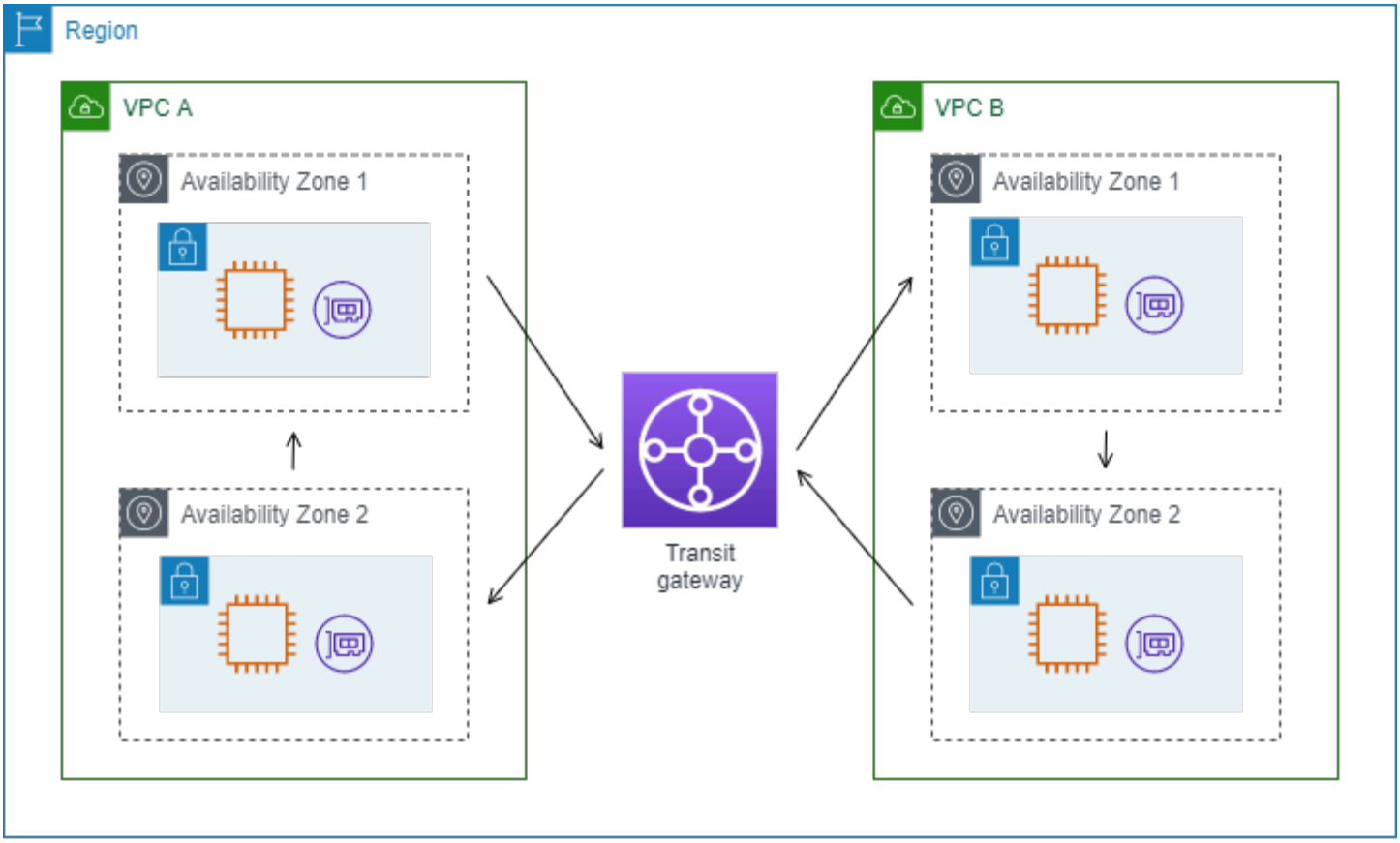
```
- eni-1235b8ca123456789 10.0.0.220 10.0.1.5 203.0.113.5 10.0.1.5
```

您可使用与以上相同的字段集创建另一个自定义流日志。您可为私有子网中的实例的网络接口创建流日志。在这种情况下，instance-id 字段返回与网络接口关联的实例的 ID，并且 dstaddr 和 pkt-dstaddr 字段与 srcaddr 和 pkt-srcaddr 字段没有不同。与 NAT 网关的网络接口不同，此网络接口不是流量的中间网络接口。

```
i-01234567890123456 eni-1111aaaa2222bbbb3 10.0.1.5 203.0.113.5 10.0.1.5 203.0.113.5
 #Traffic from the source instance to host on the internet
i-01234567890123456 eni-1111aaaa2222bbbb3 203.0.113.5 10.0.1.5 203.0.113.5 10.0.1.5
 #Response traffic from host on the internet to the source instance
```

通过中转网关的流量

在此示例中，VPC A 中的客户端通过中转网关连接到 VPC B 中的 Web 服务器。客户端和服务器处于不同的可用区中。流量使用弹性网络接口 ID 抵达 VPC B 中的服务器（在本例中，假设该 ID 为 eni-11111111111111111），使用另一个 ID（例如 eni-22222222222222222）离开 VPC B。



您可使用以下格式为 VPC B 创建自定义流日志。

```
version interface-id account-id vpc-id subnet-id instance-id srcaddr dstaddr srcport
 dstport protocol tcp-flags type pkt-srcaddr pkt-dstaddr action log-status
```

来自流日志记录的以下行演示了 Web 服务器上网络接口的流量流。第一行是来自客户端的请求流量，最后一行是来自 Web 服务器的响应流量。

```
3 eni-33333333333333333 123456789010 vpc-abcdefab012345678 subnet-22222222bbbbbbbbb
 i-01234567890123456 10.20.33.164 10.40.2.236 39812 80 6 3 IPv4 10.20.33.164
 10.40.2.236 ACCEPT OK
...
3 eni-33333333333333333 123456789010 vpc-abcdefab012345678 subnet-22222222bbbbbbbbb
 i-01234567890123456 10.40.2.236 10.20.33.164 80 39812 6 19 IPv4 10.40.2.236
 10.20.33.164 ACCEPT OK
```

以下行是 eni-11111111111111111 上的请求流量，这是子网 subnet-11111111aaaaaaaaa 中的中转网关的请求方管理的网络接口。因此，以下日志记录为 instance-id 字段显示“-”符号。srcaddr 字段显示中转网关网络接口的私有 IP 地址，pkt-srcaddr 字段显示 VPC A 中客户端的源 IP 地址。

```
3 eni-1111111111111111 123456789010 vpc-abcdefab012345678 subnet-11111111aaaaaaaaa -
 10.40.1.175 10.40.2.236 39812 80 6 3 IPv4 10.20.33.164 10.40.2.236 ACCEPT OK
```

以下行是 eni-22222222222222222 上的响应流量，这是子网 subnet-22222222bbbbbbbbb 中的中转网关的请求方管理的网络接口。dstaddr 字段显示中转网关网络接口的私有 IP 地址，pkt-dstaddr 字段显示 VPC A 中客户端的 IP 地址。

```
3 eni-22222222222222222 123456789010 vpc-abcdefab012345678 subnet-22222222bbbbbbbbb -
 10.40.2.236 10.40.2.31 80 39812 6 19 IPv4 10.40.2.236 10.20.33.164 ACCEPT OK
```

服务名称、流量路径和流向

下面是自定义流日志记录的字段示例。

```
version srcaddr dstaddr srcport dstport protocol start end type packets bytes account-
id vpc-id subnet-id instance-id interface-id region az-id sublocation-type sublocation-
id action tcp-flags pkt-srcaddr pkt-dstaddr pkt-src-aws-service pkt-dst-aws-service
 traffic-path flow-direction log-status
```

在以下示例中，版本为 5，因为记录包含版本 5 字段。EC2 实例调用 Amazon S3 服务。流日志将在实例的网络接口上捕获。第一条记录的流向为 ingress，第二条记录的流向为 egress。对于 egress 记录，traffic-path 为 8，表示流量经过互联网网关。traffic-path 流量不支持 ingress 字段。当 pkt-srcaddr 或 pkt-dstaddr 是公有 IP 地址时，将显示服务名称。

```
5 52.95.128.179 10.0.0.71 80 34210 6 1616729292 1616729349 IPv4 14 15044
 123456789012 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-0c50d5961bcb2d47b
 eni-1235b8ca123456789 ap-southeast-2 apse2-az3 - - ACCEPT 19 52.95.128.179 10.0.0.71
 S3 - - ingress OK
5 10.0.0.71 52.95.128.179 34210 80 6 1616729292 1616729349 IPv4 7 471 123456789012 vpc-
abcdefab012345678 subnet-aaaaaaaa012345678 i-0c50d5961bcb2d47b eni-1235b8ca123456789
 ap-southeast-2 apse2-az3 - - ACCEPT 3 10.0.0.71 52.95.128.179 - S3 8 egress OK
```

流日志限制

要使用流日志，您需要了解以下限制：

- 创建流日志后，在您选择的网络接口、子网或 VPC 中有活跃流量之前，将无法看到流日志数据。
- 您不能为与您的 VPC 对等的 VPC 启用流日志，除非该对等 VPC 在您的账户中。

- 创建流日志后，将无法更改其配置或者流日志记录格式。例如，您无法将不同的 IAM 角色与流日志关联，或者在流日志记录中添加或删除字段。不过，您可以删除流日志并使用必需的配置创建新的流日志。
- 如果网络接口有多个 IPv4 地址，并且流量发送到辅助私有 IPv4 地址，则流日志会在 `dstaddr` 字段中显示主要私有 IPv4 地址。要捕获原始目标 IP 地址，请使用 `pkt-dstaddr` 字段创建流日志。
- 如果流量发送到某个网络接口而目标不是网络接口 IP 地址中的任何一个，则流日志会在 `dstaddr` 字段中显示主要私有 IPv4 地址。要捕获原始目标 IP 地址，请使用 `pkt-dstaddr` 字段创建流日志。
- 如果流量是从某个网络接口发送的，并且源不是任何网络接口的 IP 地址，则当日志记录用于传出流时，流日志会在 `srcaddr` 字段中显示主要私有 IPv4 地址。要捕获原始源 IP 地址，请使用 `pkt-srcaddr` 字段创建流日志。如果日志记录用于进入网络接口的入口流，则网络接口的主要私有 IP 将不会显示在 `srcaddr` 字段中。
- 当您的网络接口附加到基于 Nitro 的实例时，无论指定的最大聚合时间间隔为多少，聚合时间间隔始终不超过 1 分钟。
- 对于 `pkt-srcaddr` 和 `pkt-dstaddr` 字段，如果中间层启用了客户端 IP 地址保留，则此字段可能会显示保留的客户端 IP，而不是中间层的 IP 地址。
- 在聚合时间间隔内可能会跳过一些流日志记录（请参阅可用字段中的 log-status）。这可能是因为存在内部 AWS 容量限制或内部错误。如果使用 AWS Cost Explorer 查看 VPC 流日志费用，并且在流日志聚合时间间隔内跳过了一些流日志，则 AWS Cost Explorer 中报告的流日志数量会高于 Amazon VPC 发布的流日志数量。
- 如果您使用 VPC 屏蔽公共访问权限（BPA）：
  - VPC BPA 的流日志不包括跳过的记录。
  - 即使您在流日志中包含 `bytes` 字段，VPC BPA 的流日志也不会包含 `bytes`。

流日志不会捕获所有 IP 流量。以下类型的流量不予以记录：

- 实例与 Amazon DNS 服务器联系时生成的流量。如果您使用自己的 DNS 服务器，则将记录到该 DNS 服务器的所有流量。
- Windows 实例为 Amazon Windows 许可证激活而生成的流量。
- 实例元数据传入和传出 `169.254.169.254` 的流量。
- Amazon Time Sync Service 的传入和传出 `169.254.169.123` 的流量。
- DHCP 流量。
- 镜像源流量的流量。您只会看到镜像目标流量的流量。

- 到默认 VPC 路由器的预留 IP 地址的流量。
- 端点网络接口和网络负载均衡器网络接口之间的流量。
- 地址解析协议（ARP）流量。

特定于版本 7 中可用的 ECS 字段的限制：

- 要创建带有 ECS 字段的流日志订阅，账户必须至少包含一个 ECS 集群。
- 如果底层 ECS 任务不属于流日志订阅的所有者，则不计算 ECS 字段。例如，如果您与其他账户（AccountB）共享子网（SubnetA），然后为 SubnetA 创建流日志订阅，则如果 AccountB 在共享子网中启动 ECS 任务，则您的订阅将收到来自 AccountB 启动的 ECS 任务的流量日志，但出于安全考虑，将不会计算这些日志的 ECS 字段。
- 如果在 VPC/子网资源级别创建带有 ECS 字段的流日志订阅，则也会为您的订阅传输为非 ECS 网络接口生成的所有流量。对于非 ECS IP 流量，ECS 字段的值将为“-”。例如，您有一个子网（subnet-000000），并且为该子网创建了带有 ECS 字段（fl-00000000）的流日志订阅。在 subnet-000000 中，您可以启动一个连接到互联网并正在积极生成 IP 流量的 EC2 实例（i-0000000）。您还可以在同一子网中启动正在运行的 ECS 任务（ECS-Task-1）。由于 i-0000000 和 ECS-Task-1 都在生成 IP 流量，因此您的流日志订阅 fl-00000000 将为两个实体提供流量日志。但是，仅 ECS-Task-1 会有您在 logFormat 中包含的 ECS 字段的实际 ECS 元数据。对于 i-0000000 相关流量，这些字段的值将为“-”。
- ecs-container-id 和 ecs-second-container-id 在 VPC 流日志服务从 ECS 事件流接收它们时进行排序。不能保证它们的顺序与您在 ECS 控制台或 DescribeTask API 调用中看到的顺序相同。如果容器在任务仍在运行时进入“已停止”状态，则它可能会继续出现在您的日志中。
- ECS 元数据和 IP 流量日志来自两个不同来源。当我们从上游依赖项中获得所有所需信息后，我们立即开始计算您的 ECS 流量。在您启动新任务后，我们将开始计算您的 ECS 字段：1) 当我们收到底层网络接口的 IP 流量时；2) 当我们收到包含您的 ECS 任务元数据的 ECS 事件以表明该任务正在运行时。在您停止任务后，我们会停止计算您的 ECS 字段：1) 当我们不再收到底层网络接口的 IP 流量或收到延迟超过一天的 IP 流量时；2) 当我们收到包含您的 ECS 任务元数据的 ECS 事件以表明您的任务不再运行时。
- 仅支持在 awsvpc 网络模式下启动的 ECS 任务。

定价

发布流日志时，将收取已出售日志的数据摄取和存档费用。有关发布已出售日志时定价的更多信息，请打开 Amazon CloudWatch 定价，选择 Logs（日志），然后找到 Vended Logs（已出售日志）。

若要跟踪发布流日志所产生的费用，您可以将成本分配标签应用到目的地资源。此后，您的 AWS 成本分配报告中就会包含按这些标签汇总的用量和成本。您可以应用代表业务类别（例如成本中心、应用程序名称或拥有者）的标签，以便整理您的成本。有关更多信息，请参阅下列内容：

- 《AWS Billing 用户指南》中的使用成本分配标签
- 《Amazon CloudWatch Logs 用户指南》中的在 Amazon CloudWatch Logs 中标记日志组
- 《Amazon Simple Storage Service 用户指南》中的使用成本分配 S3 存储桶标签
- 《Amazon Data Firehose Developer Guide》中的 Tagging Your Delivery Streams

使用流日志

您可以使用 Amazon EC2 和 Amazon VPC 的控制台处理流日志。

任务

- 1. 借助 IAM 控制对流日志的使用
- 2. 创建流日志
- 3. 标记流日志
- 4. 删除流日志
- 命令行概述

1. 借助 IAM 控制对流日志的使用

默认情况下，用户无权使用流日志。您可以创建一个 IAM 角色，并附加向该角色授予流日志创建、描述和删除权限的策略。

下面是一个示例策略，该策略向用户授予创建、描述和删除流日志的完全权限。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DeleteFlowLogs",
```

```
        "ec2:CreateFlowLogs",
        "ec2:DescribeFlowLogs"
      ],
      "Resource": "*"
    }
  ]
}
```

有关更多信息，请参阅 the section called “Amazon VPC 如何与 IAM 配合使用”。

2. 创建流日志

您可以为 VPC、子网或网络接口创建流日志。创建流日志时，您必须为流日志指定目的地。有关更多信息，请参阅下列内容：

- the section called “创建发布到 CloudWatch Logs 的流日志”
- the section called “创建发布到 Amazon S3 的流日志”
- the section called “创建发布到 Amazon Data Firehose 的流日志”

3. 标记流日志

您可以随时为流日志添加或删除标签。

管理流日志的标签

1. 请执行以下操作之一：
   - 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。在导航窗格中，选择网络接口。选中该网络接口的复选框。
   - 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。在导航窗格中，选择 Your VPCs(您的 VPC)。选中该 VPC 的复选框。
   - 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。在导航窗格中，选择 Subnets(子网)。选中子网的复选框。
2. 选择 Flow Logs（流日志）。
3. 依次选择 Actions（操作）、Manage tags（管理标签）。
4. 若要添加新标签，请选择 Add new tag（添加新标签），然后输入键和值。要删除标签，请选择移除。
5. 添加完或删除完标签后，选择 Save（保存）。

4. 删除流日志

您可以随时删除流日志。删除流日志之后，可能需要几分钟时间才能停止收集数据。

删除流日志不会从目的地中删除日志数据，也不会修改目的地资源。您必须使用目的地服务控制台，从目的地中直接删除现有流日志数据并清理目的地资源。

删除流日志

1. 请执行以下操作之一：
   - 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。在导航窗格中，选择网络接口。选中该网络接口的复选框。
   - 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。在导航窗格中，选择 Your VPCs(您的 VPC)。选中该 VPC 的复选框。
   - 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。在导航窗格中，选择 Subnets(子网)。选中子网的复选框。
2. 选择 Flow Logs（流日志）。
3. 依次选择 Actions（操作）、Delete flow logs（删除流日志）。
4. 提示进行确认时，键入 **delete**，然后选择 Delete（删除）。

命令行概述

您可以使用命令行执行此页面上介绍的任务。

创建流日志

- create-flow-logs（AWS CLI）
- New-EC2FlowLog（AWS Tools for Windows PowerShell）

描述流日志

- describe-flow-logs（AWS CLI）
- Get-EC2FlowLog（AWS Tools for Windows PowerShell）

标记流日志

- create-tags 和 delete-tags (AWS CLI)

- New-EC2Tag 和 Remove-EC2Tag（AWS Tools for Windows PowerShell）

删除流日志

- delete-flow-logs（AWS CLI）
- Remove-EC2FlowLog（AWS Tools for Windows PowerShell）

将流日志发布到 CloudWatch Logs

流日志可以将流日志数据直接发布到 Amazon CloudWatch。Amazon CloudWatch 是一项全面的监控和可观测性服务。该服务会收集并跟踪来自多种 AWS 资源以及您自有应用程序和服务的指标、日志和事件数据。CloudWatch 可让您了解资源利用率、应用程序性能和运行状况，让您能够检测并响应系统范围的性能变化和潜在问题。您可以利用 CloudWatch 来设置警报、可视化日志和指标，并自动做出反应来收集并优化自己的云资源。CloudWatch 是保障基于云的基础设施与应用程序的可靠性、可用性和性能的重要工具。

在发布到 CloudWatch Logs 时，流日志数据将发布到日志组，并且每个网络接口在该日志组中有唯一的日志流。日志流包含流日志记录。您可以创建将数据发布到相同日志组的多个流日志。如果相同日志组中的一个或多个流日志存在相同网络接口，其中就会有一个组合日志流。如果您指定了一个流日志应该捕获已拒绝流量，而另一个流日志应该捕获已接受流量，则组合日志流会捕获所有流量。

在 CloudWatch Logs 中，timestamp (时间戳) 字段对应于流日志记录中捕获的开始时间。ingestionTime 字段指示 CloudWatch Logs 开始接收流日志记录的日期和时间。此时间戳晚于在流日志记录中捕获的结束时间。

有关 CloudWatch Logs 的更多信息，请参阅 Amazon CloudWatch Logs 用户指南中的发送到 CloudWatch Logs 的日志。

定价

将流日志发布到 CloudWatch Logs 时，适用已出售日志的数据引入和存档费用。有关更多信息，请打开 Amazon CloudWatch Pricing（Amazon CloudWatch 定价），选择 Logs（日志），找到 Vended Logs（已出售日志）。

内容

- 用于将流日志发布到 CloudWatch Logs 的 IAM 角色
- 创建发布到 CloudWatch Logs 的流日志

- 借助 CloudWatch Logs 查看流日志记录
- 搜索流日志记录
- 处理 CloudWatch Logs 中的流日志记录

用于将流日志发布到 CloudWatch Logs 的 IAM 角色

与您的流日志关联的 IAM 角色必须具有足够的权限，以便将流日志发布到 CloudWatch Logs 中的指定日志组。IAM 角色必须属于您的 AWS 账户。

附加到您的 IAM 角色的 IAM 策略必须至少包括以下权限。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "logs:DescribeLogGroups",
        "logs:DescribeLogStreams"
      ],
      "Resource": "*"
    }
  ]
}
```

确保您的角色具有以下信任策略，允许流日志服务代入该角色。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```
      "Effect": "Allow",
      "Principal": {
        "Service": "vpc-flow-logs.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

建议您使用 aws:SourceAccount 和 aws:SourceArn 条件键来防止出现混淆代理人问题。例如，您可以将以下条件块添加到以前的信任策略。源账户是流日志的所有者，并且源 ARN 是流日志 ARN。如果您不知道流日志 ID，则可以用通配符（*）替换 ARN 的该部分，然后在创建流日志后更新策略。

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account_id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:ec2:region:account_id:vpc-flow-log/flow-log-id"
    }
}
```

为流日志创建 IAM 角色

如上所述，您可以更新现有角色。或者，您可以使用以下步骤创建用于流日志的新角色。您将在创建流日志时指定该角色。

为流日志创建 IAM 角色

1. 打开 IAM 控制台：https://console.aws.amazon.com/iam/。
2. 在导航窗格中，选择策略。
3. 选择创建策略。
4. 在创建策略页面上，执行以下操作：
   a. 选择 JSON。
   b. 将此窗口的内容替换为此部分开头的权限策略。
   c. 选择下一步。
   d. 输入您的策略名称以及可选的描述和标签，然后选择创建策略。

5. 在导航窗格中，选择角色。
6. 选择创建角色。
7. 对于 Trusted entity type（可信实体类型），选择 Custom trust policy（自定义信任策略）。对于 Custom trust policy（自定义信任策略），将 `"Principal": {},` 替换为以下内容，然后选择 Next（下一步）。

```
"Principal": {
   "Service": "vpc-flow-logs.amazonaws.com"
},
```

8. 在 Add permissions（添加权限）页面上，选中您在此过程中先前创建的策略复选框，然后选择 Next（下一步）。
9. 输入您的角色的名称，并且可以选择提供描述。
10. 选择Create role（创建角色）。

创建发布到 CloudWatch Logs 的流日志

您可以为 VPC、子网或网络接口创建流日志。如果以使用特定 IAM 角色的用户身份执行这些步骤，请确保该角色具有使用 `iam:PassRole` 操作的权限。

先决条件

验证用于发出请求的 IAM 主体是否具有调用 `iam:PassRole` 操作的权限。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::111122223333:role/flow-log-role-name"
        }
    ]
}
```

使用控制台创建流日志

1. 请执行以下操作之一：
   - 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。在导航窗格中，选择网络接口。选中该网络接口的复选框。
   - 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。在导航窗格中，选择 Your VPCs(您的 VPC)。选中该 VPC 的复选框。
   - 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。在导航窗格中，选择 Subnets(子网)。选中子网的复选框。
2. 选择 Actions（操作）、Create flow log（创建流日志）。
3. 对于Filter（筛选条件），指定要记录的流量的类型。选择All（全部）将记录接受和拒绝的流量，选择 Reject（拒绝）将仅记录被拒绝的流量，选择 Accept（接受）将仅记录接受的流量。
4. 对于 Maximum aggregation interval（最大聚合时间间隔），选择捕获流并聚合到一个流日志记录中的最大时间段。
5. 对于Destination（目的地），选择发送到 CloudWatch Logs。
6. 对于目标日志组，选择现有日志组名称或输入新日志组名称。如果您输入名称，我们会在需要记录流量时创建日志组。
7. 对于访问服务，请选择有权将日志发布到 CloudWatch Logs 的现有 IAM 服务角色，或者选择创建新的服务角色。
8. 对于Log record format（日志记录格式），选定流日志记录的格式。
   - 要使用默认格式，请选择AWSdefault format（亚马逊云科技默认格式）。
   - 要使用自定义格式，请选择Custom format（自定义格式）然后从Log format（日志格式）选择字段。
9. 对于其他元数据，选择是否要以日志格式包含来自 Amazon ECS 的元数据。
10. （可选）选择Add new tag（添加新标签）以将标签应用于流日志。
11. 选择 Create flow log（创建流日志）。

使用命令行创建流日志

使用以下命令之一。

- create-flow-logs（AWS CLI）
- New-EC2FlowLog（AWS Tools for Windows PowerShell）

以下 AWS CLI 示例将创建流日志，该日志将捕获指定子网的所有已接受流量。流日志将传输到指定的日志组。`--deliver-logs-permission-arn` 参数指定发布到 CloudWatch Logs 所需的 IAM 角色。

```
aws ec2 create-flow-logs --resource-type Subnet --resource-ids subnet-1a2b3c4d --
traffic-type ACCEPT --log-group-name my-flow-logs --deliver-logs-permission-arn
 arn:aws:iam::123456789101:role/publishFlowLogs
```

借助 CloudWatch Logs 查看流日志记录

您可以使用 CloudWatch Logs 控制台查看流日志记录。创建流日志之后，可能需要几分钟才能在控制台中显示。

查看使用控制台发布到 CloudWatch Logs 的流日志记录

1. 通过 https://console.aws.amazon.com/cloudwatch/ 打开 CloudWatch 控制台。
2. 在导航窗格中，依次选择 Logs（日志）和 Log groups（日志组）。
3. 选择包含流日志的日志组名称以打开其详细信息页面。
4. 选择包含流日志记录的日志流名称。有关更多信息，请参阅 流日志记录。

使用命令行查看发布到 CloudWatch Logs 的流日志记录

- get-log-events（AWS CLI）
- Get-CWLLogEvent（AWS Tools for Windows PowerShell）

搜索流日志记录

您可以使用 CloudWatch Logs 控制台搜索发布到 CloudWatch Logs 的流日志记录。您可以使用度量筛选器筛选流日志记录。流日志记录用空格分隔。

使用 CloudWatch Logs 控制台搜索流日志记录

1. 通过以下网址打开 CloudWatch 控制台：https://console.aws.amazon.com/cloudwatch/。
2. 在导航窗格中，依次选择 Logs（日志）和 Log groups（日志组）。
3. 如果您知道要搜索的网络接口，请选择包含流日志的日志组，然后选择日志流。或者，选择 Search log group（搜索日志组）。如果日志组中有许多网络接口，或者根据您选择的时间范围，这可能需要一些时间。

4. 在筛选事件下，输入以下字符串。这假定流日志记录使用默认格式。

```
[version, accountid, interfaceid, srcaddr, dstaddr, srcport, dstport, protocol,
 packets, bytes, start, end, action, logstatus]
```

5. 通过为字段指定值，根据需要修改筛选器。以下示例按特定的源 IP 地址进行筛选。

```
[version, accountid, interfaceid, srcaddr = 10.0.0.1, dstaddr, srcport, dstport,
 protocol, packets, bytes, start, end, action, logstatus]
[version, accountid, interfaceid, srcaddr = 10.0.2.*, dstaddr, srcport, dstport,
 protocol, packets, bytes, start, end, action, logstatus]
```

以下示例按目标端口、字节数以及流量是否被拒绝进行筛选。

```
[version, accountid, interfaceid, srcaddr, dstaddr, srcport, dstport = 80 ||
 dstport = 8080, protocol, packets, bytes, start, end, action, logstatus]
[version, accountid, interfaceid, srcaddr, dstaddr, srcport, dstport = 80 ||
 dstport = 8080, protocol, packets, bytes >= 400, start, end, action = REJECT,
 logstatus]
```

处理 CloudWatch Logs 中的流日志记录

您可以像处理由 CloudWatch Logs 收集的其余日志事件一样处理流日志记录。有关监控日志数据和指标筛选条件的更多信息，请参阅《Amazon CloudWatch Logs 用户指南》中的 Creating metrics from log events using filter。

示例：为流日志创建 CloudWatch 指标筛选条件和警报

在此示例中，您有一个适用于 eni-1a2b3c4d 的流日志。您要创建一个警报，如果 1 小时内有 10 次或超过 10 次通过 TCP 端口 22（SSH）连接到您的实例的尝试遭到拒绝，该警报将向您发出提醒。首先，您必须创建一个指标筛选条件，该指标筛选条件与为其创建警报的流量的模式相匹配。然后，您可以为该指标筛选条件创建警报。

为已拒绝的 SSH 流量创建指标筛选条件并为该筛选条件创建警报

1. 通过以下网址打开 CloudWatch 控制台：https://console.aws.amazon.com/cloudwatch/。
2. 在导航窗格中，依次选择 Logs（日志）和 Log groups（日志组）。
3. 选中日志组对应的复选框，然后选择 Actions（操作）、Create metric filter（创建指标筛选条件）。

4. 对于 Filter Pattern（筛选条件模式），输入以下字符串。

```
[version, account, eni, source, destination, srcport, destport="22", protocol="6",
 packets, bytes, windowstart, windowend, action="REJECT", flowlogstatus]
```

5. 对于 Select log data to test（选择要测试的日志数据），选择您的网络接口对应的日志流。（可选）要查看与筛选条件模式匹配的日志数据行，请选择 Test pattern（测试模式）。
6. 准备就绪后，选择 Next（下一步）。
7. 输入筛选条件名称、指标命名空间和指标名称。将指标值设置为 1。完成后，选择 Next（下一步），然后选择 Create metric filter（创建指标筛选条件）。
8. 在导航窗格中，依次选择 Alarms（警报）和 All alarms（所有警报）。
9. 选择Create alarm（创建警报）。
10. 选择您创建的指标名称，然后选择选择指标。
11. 按如下所示配置警报，然后选择 Next（下一步）：
    - 对于 Statistic（统计数据），选择 Sum（总计）。这可以确保您捕获指定时间段内的数据点的总数。
    - 对于 Period（周期），选择 1 hour（1 小时）。
    - 对于每当 TimeSinceLastActive...，选择大于/等于，然后输入 10 作为阈值。
    - 对于 Additional configuration（其他配置）、Datapoints to alarm（警报的数据点数），将默认值设为 1。
12. 选择下一步。
13. 对于 Notification（通知），选择现有的 SNS 主题，或选择 Create new topic（新建主题）创建一个新主题。选择 Next（下一步）。
14. 输入警报的名称和描述，然后选择 Next（下一步）。
15. 预览完警报后，选择创建警报。

将流日志发布到 Amazon S3

流日志可以将流日志数据发布到 Amazon S3。Amazon S3（Simple Storage Service）是一种高度可扩展和持久的对象存储服务。该服务专为在 Web 的任何位置存储和检索任意数量的数据而构建。S3 具有业界领先的持久性和可用性，拥有内置的数据版本控制、加密和访问控制等功能。

在发布到 Amazon S3 时，流日志数据将发布到您指定的现有 Amazon S3 存储桶。所有受监控网络接口的流日志记录将发布到存储桶中存储的一系列日志文件对象。如果流日志捕获 VPC 的数据，流日志将发布选定 VPC 中所有网络接口的流日志记录。

要创建用于流日志的 Amazon S3 存储桶，请参阅《Amazon S3 用户指南》中的创建桶。

有关如何简化 VPC 流日志摄取、流日志处理和流日志可视化的更多信息，请参阅 AWS 解决方案库中的 Centralized Logging with OpenSearch。

有关 CloudWatch Logs 的更多信息，请参阅《Amazon CloudWatch Logs 用户指南》中的发送到 Amazon S3 的日志。

定价

将流日志发布到 Amazon S3 时，适用已出售日志的数据引入和存档费用。有关更多信息，请打开 Amazon CloudWatch Pricing（Amazon CloudWatch 定价），选择 Logs（日志），找到 Vended Logs（已出售日志）。

内容

- 流日志文件
- 针对流日志的 Amazon S3 存储桶权限
- 与 SSE-KMS 结合使用时必需的密钥策略
- Amazon S3 日志文件权限
- 创建发布到 Amazon S3 的流日志
- 借助 Amazon S3 查看流日志记录

流日志文件

VPC 流日志将进出您的 VPC 的 IP 流量的相关数据收集到日志记录，并将这些记录聚合到日志文件，然后每隔 5 分钟将日志文件发布到 Amazon S3 存储桶。可能会发布多个文件，并且每个日志文件可能包含在上一个 5 分钟内记录的 IP 流量的部分或全部流日志记录。

在 Amazon S3 中，流日志文件的 Last modified（上次修改时间）字段指示文件上传到 Amazon S3 存储桶的日期和时间。此时间要晚于文件名中的时间戳，并且不同于将文件上传到 Amazon S3 存储桶所花费的时间。

日志文件格式

您可为日志文件指定下列格式之一。每个文件都被压缩为单个 Gzip 文件。

- Text – 纯文本。这是默认格式。
- Parquet – Apache Pparquet 是一种列式数据格式。与对纯文本数据的查询相比，对 Passic 格式的数据进行查询速度快 10 到 100 倍。使用 Gzip 压缩的 Parquet 格式的数据比 Gzip 压缩的纯文本格式的数据占用的存储空间少 20%。

> **Note**
>
> 如果采用 Gzip 压缩的 Parquet 格式的数据在每个聚合周期内小于 100 KB，则由于 Parquet 文件内存要求，以 Parquet 格式存储数据可能比采用 Gzip 压缩的纯文本占用更多的空间。

日志文件选项

您也可以指定以下选项。

- Hive 兼容的 S3 前缀 – 启用 Hive兼容的前缀，而不是将分区导入 Hive 兼容工具中。请先使用 MSCK REPAIR TABLE 命令，然后再运行查询。
- 每小时分区 – 如果您有大量日志并且通常将查询定位到特定小时，则可以通过每小时对日志进行分区来获得更快的结果并节省查询成本。

日志文件 S3 存储桶结构

日志文件将保存到指定的 Amazon S3 存储桶，并使用由流日志的 ID、区域、创建日期及目标选项决定的文件夹结构。

默认情况下，文件传送到以下位置。

```
bucket-and-optional-prefix/AWSLogs/account_id/vpcflowlogs/region/year/month/day/
```

如果启用 Hive 兼容的 S3 前缀，则文件将传送到以下位置。

```
bucket-and-optional-prefix/AWSLogs/aws-account-id=account_id/aws-service=vpcflowlogs/
aws-region=region/year=year/month=month/day=day/
```

如果启用每小时分区，则文件将传送到以下位置。

```
bucket-and-optional-prefix/AWSLogs/account_id/vpcflowlogs/region/year/month/day/hour/
```

如果启用 Hive 兼容的分区并每小时对流日志进行分区，则文件将传送到以下位置。

```
bucket-and-optional-prefix/AWSLogs/aws-account-id=account_id/aws-service=vpcflowlogs/
aws-region=region/year=year/month=month/day=day/hour=hour/
```

日志文件名称

日志文件的文件名基于流日志 ID、区域以及创建日期和时间。文件名使用以下格式。

```
aws_account_id_vpcflowlogs_region_flow_log_id_YYYYMMDDTHHmmZ_hash.log.gz
```

下面显示了一个流日志的日志文件的示例，该流日志由 AWS 账户 123456789012 创建，用于 us-east-1 区域中的资源，创建时间为 June 20, 2018 16:20 UTC。该文件包含结束时间介于 16:20:00 和 16:24:59 之间的流日志记录。

```
123456789012_vpcflowlogs_us-east-1_fl-1234abcd_20180620T1620Z_fe123456.log.gz
```

针对流日志的 Amazon S3 存储桶权限

默认情况下，Amazon S3 存储桶以及其中包含的对象都是私有的。只有存储桶拥有者才能访问存储桶和其中存储的对象。不过，存储桶拥有者可以通过编写访问策略来向其他资源和用户授予访问权限。

如果创建流日志的用户拥有存储桶并且对它具有 PutBucketPolicy 和 GetBucketPolicy 权限，则我们会自动将以下策略附加到存储桶。此策略将覆盖附加到存储桶的任何现有策略。

否则，存储桶拥有者必须将此策略添加到存储桶中，以指定流日志创建者的 AWS 账户 ID，否则流日志创建失败。有关更多信息，请参阅《Amazon Simple Storage Service 用户指南》中的使用存储桶策略。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AWSLogDeliveryWrite",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
```

```
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "123456789012",
                    "s3:x-amz-acl": "bucket-owner-full-control"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:logs:us-east-1:123456789012:*"
                }
            }
        },
        {
            "Sid": "AWSLogDeliveryAclCheck",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "123456789012"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:logs:us-east-1:123456789012:*"
                }
            }
        }
    ]
}
```

您为 *my-s3-arn* 指定的 ARN 取决于您是否使用 Hive 兼容的 S3 前缀。

- 默认前缀

```
arn:aws:s3:::bucket_name/optional_folder/AWSLogs/account_id/*
```

- Hive 兼容的 S3 前缀

```
arn:aws:s3:::bucket_name/optional_folder/AWSLogs/aws-account-id=account_id/*
```

最佳实践是向日志传输服务主体（而不是单个 AWS 账户 ARN）授予这些权限。此外，最好是使用 `aws:SourceAccount` 和 `aws:SourceArn` 条件键来防止出现混淆代理人问题。源账户是流日志的所有者，并且源 ARN 是日志服务的通配符（*）ARN。

请注意，日志传输服务调用 `HeadBucket` Amazon S3 API 操作来验证 S3 存储桶的存在及位置。您无需授予日志传输服务调用此操作的权限；即使它无法确认 S3 存储桶的存在及其位置，仍会传输 VPC 流日志。但是，CloudTrail 日志中会出现 `HeadBucket` 调用 `AccessDenied` 错误。

与 SSE-KMS 结合使用时必需的密钥策略

您可以通过启用 Amazon S3 托管式密钥的服务器端加密 (SSE-S3) 或 S3 存储桶上 KMS 密钥（SSE-KMS）的服务器端加密 (SSE-KMS) 来保护 Amazon S3 存储桶中的数据。有关详情，请参阅《Amazon S3 用户指南》中的使用服务器端加密保护数据。

如果选择 SSE-S3，则不需要额外的配置。Amazon S3 处理加密密钥。

如果您选择 SSE-KMS，则必须使用客户自主管理型密钥 ARN。如果您使用密钥 ID，则在创建流日志时可能会遇到 LogDestination 无法送达 错误。此外，您还必须更新客户自主管理型密钥的密钥政策，以确保日志传输账户可以写入您的 S3 存储桶。有关与 SSE-KMS 结合使用时必需的密钥策略的更多信息，请参阅 Amazon CloudWatch Logs 用户指南中的 Amazon S3 存储桶服务器端加密。

Amazon S3 日志文件权限

除了必需的存储桶策略之外，Amazon S3 使用访问控制列表（ACL）管理对流日志创建的日志文件的访问。默认情况下，存储桶拥有者对每个日志文件具有 `FULL_CONTROL` 权限。如果日志传输拥有者与存储桶拥有者不同，则没有权限。日志传输账户具有 `READ` 和 `WRITE` 权限。有关更多信息，请参阅《Amazon S3 用户指南》中的访问控制列表 (ACL) 概述。

创建发布到 Amazon S3 的流日志

在您创建和配置 Amazon S3 存储桶后，您可以为网络接口、子网和 VPC 创建流日志。

先决条件

创建流日志的 IAM 主体必须使用具有以下权限的 IAM 角色，才能将流日志发布到目标 Amazon S3 存储桶。

JSON

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogDelivery",
        "logs:DeleteLogDelivery"
      ],
      "Resource": "*"
    }
  ]
}
```

使用控制台创建流日志

1. 请执行以下操作之一：
   - 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。在导航窗格中，选择网络接口。选中该网络接口的复选框。
   - 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。在导航窗格中，选择 Your VPCs(您的 VPC)。选中该 VPC 的复选框。
   - 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。在导航窗格中，选择 Subnets(子网)。选中子网的复选框。
2. 选择 Actions（操作）、Create flow log（创建流日志）。
3. 对于 Filter（筛选条件），指定要记录的 IP 流量数据的类型。
   - 已接受 – 仅记录接受的流量。
   - 已拒绝 – 仅记录拒绝的流量。
   - All（所有流量）– 记录接受的和拒绝的流量。
4. 对于 Maximum aggregation interval（最大聚合时间间隔），选择捕获流并聚合到一个流日志记录中的最大时间段。
5. 对于 Destination (目标)，选择 Send to an Amazon S3 bucket (发送到 Amazon S3 存储桶)。
6. 对于 S3 bucket ARN（S3 存储桶 ARN），指定某个现有 Amazon S3 存储桶的 Amazon Resource Name（ARN）。您可以选择包含子文件夹。例如，要指定名为 `my-logs` 的存储桶中名为 `my-bucket` 的子文件夹，请使用以下 ARN：

   ```
   arn:aws:s3:::my-bucket/my-logs/
   ```

   存储桶不能使用 `AWSLogs` 作为子文件夹名称，因为这是保留项。

如果您拥有该存储桶，我们会自动创建资源策略并将它附加到该存储桶。有关更多信息，请参阅针对流日志的 Amazon S3 存储桶权限。

7. 对于 Log record format（日志记录格式），选定流日志记录的格式。
   - 要使用默认流日志记录格式，请选择 AWS default format（亚马逊云科技默认格式）。
   - 要创建自定义格式，请选择Custom format（自定义格式）。对于Log format（日志行格式），选择要包括在流日志记录中的字段。
8. 对于其他元数据，选择是否要以日志格式包含来自 Amazon ECS 的元数据。
9. 对于 Log file format（日志文件格式），指定日志文件的格式。
   - Text – 纯文本。这是默认格式。
   - Parquet – Apache Pparquet 是一种列式数据格式。与对纯文本数据的查询相比，对 Passic 格式的数据进行查询速度快 10 到 100 倍。使用 Gzip 压缩的 Parquet 格式的数据比 Gzip 压缩的纯文本格式的数据占用的存储空间少 20%。
10. （可选）要使用 Hive 兼容的 S3 前缀，请选择 Hive-compatible S3 prefix（Hive 兼容的 S3 前缀）、Enable（启用）。
11. （可选）要每小时对流日志进行分区，请选择 Every 1 hour（60 mins）（每 1 小时（60 分钟））。
12. （可选）要向流日志添加标签，请选择 Add new tag（添加新标签）并指定标签键和值。
13. 选择 Create flow log（创建流日志）。

使用命令行创建发布到 Amazon S3 的流日志

使用以下命令之一：

- create-flow-logs（AWS CLI）
- New-EC2FlowLog（AWS Tools for Windows PowerShell）

以下 AWS CLI 示例将创建流日志，以捕获指定 VPC 的所有流量并将流日志传输到指定 Amazon S3 存储桶。`--log-format` 参数指定流日志记录的自定义格式。

```
aws ec2 create-flow-logs --resource-type VPC --resource-ids vpc-00112233344556677 --
traffic-type ALL --log-destination-type s3 --log-destination arn:aws:s3:::flow-log-
bucket/custom-flow-logs/ --log-format '${version} ${vpc-id} ${subnet-id} ${instance-
```

```
id} ${srcaddr} ${dstaddr} ${srcport} ${dstport} ${protocol} ${tcp-flags} ${type} ${pkt-
srcaddr} ${pkt-dstaddr}'
```

借助 Amazon S3 查看流日志记录

您可以使用 Amazon S3 控制台查看流日志记录。创建流日志之后，可能需要几分钟才能在控制台中显示。

日志文件是压缩文件。如果您使用 Amazon S3 控制台打开这些日志文件，则将对其进行解压缩，并且将显示流日志记录。如果您下载这些文件，则必须对其进行解压才能查看流日志记录。

查看发布到 Amazon S3 的流日志记录

1. 通过以下网址打开 Amazon S3 控制台：https://console.aws.amazon.com/s3/。
2. 选择存储桶的名称以打开其详细信息页面。
3. 导航到包含日志文件的文件夹。例如，*prefix*/AWSLogs/*account_id*/vpcflowlogs/*region*/*year*/*month*/*day*/。
4. 选中文件名旁边的复选框，然后选择 Download（下载）。

您还可以使用 Amazon Athena 查询日志文件中的流日志记录。Amazon Athena 是一种交互式查询服务，让您能够更轻松地使用标准 SQL 分析 Amazon S3 中的数据。有关更多信息，请参阅 Amazon Athena 用户指南 中的查询 Amazon VPC 流日志。

将流日志发布到 Amazon Data Firehose

流日志可以将流日志数据直接发布到 Amazon Data Firehose。Amazon Data Firehose 是一项完全托管的服务，可收集、转换实时数据流并将此类数据流传送到各种 AWS 数据存储和分析服务中。该服务可代表您处理数据摄取任务。

就 VPC 流日志而言，Firehose 非常适用。VPC 流日志可捕获有关在 VPC 中传入和传出网络接口的 IP 流量的信息。此类数据对于安全监控、性能分析和监管合规性至关重要。不过，管理这种持续日志数据流的存储和处理可能是一项复杂的资源密集型任务。

通过将 Firehose 与 VPC 流日志集成，您可以将此类数据传输到首选目标，例如 Amazon S3、Amazon Redshift 或 Amazon OpenSearch Service。Firehose 会就处理 VPC 流日志摄取、转换和传输等任务进行扩展，从而减轻您的运营负担，让您可以专注于分析日志和获取洞察，而不必担心底层基础设施。

此外，Firehose 还提供数据转换、压缩和加密等功能，可以提高 VPC 流日志处理管道的效率和安全性。将 Firehose 用于 VPC 流日志可以简化数据管理工作，让您能够从网络流量数据中获得洞察。

流日志数据发布到 Amazon Data Firehose 时，会以纯文本格式发布到 Amazon Data Firehose 传输流。

定价

将收取标准摄取和传输费用。有关更多信息，请打开 Amazon CloudWatch Pricing（Amazon CloudWatch 定价），选择 Logs（日志），找到 Vended Logs（已出售日志）。

内容

- 用于跨账户传输的 IAM 角色
- 创建发布到 Amazon Data Firehose 的流日志

用于跨账户传输的 IAM 角色

发布到 Amazon Data Firehose 时，您可以选择与要监控的资源位于同一账户（源账户）或不同账户（目的地账户）中的传输流。要启用跨账户将流日志传输到 Amazon Data Firehose，您必须在源账户中创建 IAM 角色，并在目的地账户中创建 IAM 角色。

角色

- 源账户角色
- 目的地账户角色

源账户角色

在源账户中，创建授予以下权限的角色。在此示例中，角色的名称为 `mySourceRole`，但您也可以为该角色选择其他名称。最后一条语句允许目的地账户中的角色代入该角色。条件语句确保该角色仅传递给日志传输服务，并且仅在监控指定资源时传递。创建策略时，使用条件键 `iam:AssociatedResourceARN` 指定要监控的 VPC、网络接口或子网。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::123456789012:role/mySourceRole",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "delivery.logs.amazonaws.com"
                },
                "StringLike": {
                    "iam:AssociatedResourceARN": [
                        "arn:aws:ec2:us-
east-1:123456789012:vpc/vpc-00112233344556677"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:ListLogDeliveries",
                "logs:GetLogDelivery"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "sts:AssumeRole",
            "Resource": "arn:aws:iam::111122223333:role/
AWSLogDeliveryFirehoseCrossAccountRole"
        }
    ]
}
```

确保该角色具有以下信任策略，允许日志传输服务代入该角色。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "delivery.logs.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

使用以下步骤从源账户中创建角色。

创建源账户角色

1. 打开 IAM 控制台：https://console.aws.amazon.com/iam/。
2. 在导航窗格中，选择策略。
3. 选择创建策略。
4. 在创建策略页面上，执行以下操作：
   a. 选择 JSON。
   b. 将此窗口的内容替换为此部分开头的权限策略。
   c. 选择下一步。
   d. 输入您的策略名称以及可选的描述和标签，然后选择创建策略。
5. 在导航窗格中，选择角色。
6. 选择创建角色。
7. 对于 Trusted entity type（可信实体类型），选择 Custom trust policy（自定义信任策略）。对于 Custom trust policy（自定义信任策略），将 `"Principal": {},` 替换为以下内容，以指定日志传输服务。选择下一步。

```
"Principal": {
   "Service": "delivery.logs.amazonaws.com"
},
```

8. 在 Add permissions（添加权限）页面上，选中您在此过程中先前创建的策略复选框，然后选择 Next（下一步）。
9. 输入您的角色的名称，并且可以选择提供描述。
10. 选择Create role（创建角色）。

目的地账户角色

在目的地账户中，创建名称以 AWSLogDeliveryFirehoseCrossAccountRole 开头的角色。该角色必须授予以下权限。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
          "iam:CreateServiceLinkedRole",
          "firehose:TagDeliveryStream"
      ],
      "Resource": "*"
    }
  ]
}
```

确保该角色具有以下信任策略，允许您在源账户中创建的角色代入该角色。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/mySourceRole"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

使用以下步骤从目的地账户中创建角色。

创建目的地账户角色

1. 打开 IAM 控制台：https://console.aws.amazon.com/iam/。
2. 在导航窗格中，选择策略。
3. 选择创建策略。
4. 在创建策略页面上，执行以下操作：
   a. 选择 JSON。
   b. 将此窗口的内容替换为此部分开头的权限策略。
   c. 选择下一步。
   d. 输入以 AWSLogDeliveryFirehoseCrossAccountRole 为开头的策略名称，然后选择 Create policy（创建策略）。
5. 在导航窗格中，选择角色。
6. 选择创建角色。
7. 对于 Trusted entity type（可信实体类型），选择 Custom trust policy（自定义信任策略）。对于 Custom trust policy（自定义信任策略），将 `"Principal": {},` 替换为以下内容，以指定源账户角色。选择下一步。

```
"Principal": {
   "AWS": "arn:aws:iam::source-account:role/mySourceRole"
},
```

8. 在 Add permissions（添加权限）页面上，选中您在此过程中先前创建的策略复选框，然后选择 Next（下一步）。
9. 输入您的角色的名称，并且可以选择提供描述。
10. 选择Create role（创建角色）。

创建发布到 Amazon Data Firehose 的流日志

您可以为 VPC、子网或网络接口创建流日志。

先决条件

- 创建目的地 Amazon Data Firehose 传输流。使用 Direct Put 作为来源。有关更多信息，请参阅创建 Amazon Data Firehose 传输流。

- 要将流日志发布到不同账户，请创建所需的 IAM 角色，如 the section called “用于跨账户传输的 IAM 角色” 中所述。

创建发布到 Amazon Data Firehose 的流日志

1. 请执行以下操作之一：
   - 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。在导航窗格中，选择网络接口。选中该网络接口的复选框。
   - 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。在导航窗格中，选择 Your VPCs(您的 VPC)。选中该 VPC 的复选框。
   - 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。在导航窗格中，选择 Subnets(子网)。选中子网的复选框。
2. 选择 Actions（操作）、Create flow log（创建流日志）。
3. 对于Filter（筛选条件），指定要记录的流量的类型。
   - Accepted（已接受）– 仅记录接受的流量
   - Rejected（已拒绝）– 仅记录拒绝的流量
   - All（所有流量）– 记录接受和拒绝的流量
4. 对于 Maximum aggregation interval（最大聚合时间间隔），选择捕获流并聚合到一个流日志记录中的最大时间段。
5. 对于 Destination（目的地），请选择下列选项之一：
   - 发送到同一个账户中的 Amazon Data Firehose – 传输流和要监控的资源位于同一账户中。
   - 发送到不同账户中的 Amazon Data Firehose – 传输流和要监控的资源位于不同账户中。
6. 对于 Amazon Data Firehose 流名称，选择您创建的传输流。
7. [仅限跨账户传输] 对于服务访问，请选择具有发布日志权限的现有 IAM 服务角色进行跨账户传输，或者选择设置权限以打开 IAM 控制台并创建服务角色。
8. 对于 Log record format（日志记录格式），选定流日志记录的格式。
   - 要使用默认流日志记录格式，请选择 AWS default format（亚马逊云科技默认格式）。
   - 要创建自定义格式，请选择Custom format（自定义格式）。对于Log format（日志行格式），选择要包括在流日志记录中的字段。
9. 对于其他元数据，选择是否要以日志格式包含来自 Amazon ECS 的元数据。
10. （可选）选择添加标签以将标签应用于流日志。

11. 选择 Create flow log（创建流日志）。

使用命令行创建发布到 Amazon Data Firehose 的流日志

使用以下命令之一：

- create-flow-logs（AWS CLI）
- New-EC2FlowLog（AWS Tools for Windows PowerShell）

以下 AWS CLI 示例将创建流日志，用于捕获指定 VPC 的所有流量，并将流日志传输到同一账户中的指定 Amazon Data Firehose 传输流。

```
aws ec2 create-flow-logs --traffic-type ALL \
  --resource-type VPC \
  --resource-ids vpc-00112233344556677 \
  --log-destination-type kinesis-data-firehose \
  --log-destination arn:aws:firehose:us-
east-1:123456789012:deliverystream/flowlogs_stream
```

以下 AWS CLI 示例将创建流日志，用于捕获指定 VPC 的所有流量，并将流日志传输到不同账户中的指定 Amazon Data Firehose 传输流。

```
aws ec2 create-flow-logs --traffic-type ALL \
  --resource-type VPC \
  --resource-ids vpc-00112233344556677 \
  --log-destination-type kinesis-data-firehose \
  --log-destination arn:aws:firehose:us-
east-1:123456789012:deliverystream/flowlogs_stream \
  --deliver-logs-permission-arn arn:aws:iam::source-account:role/mySourceRole \
  --deliver-cross-account-role arn:aws:iam::destination-account:role/
AWSLogDeliveryFirehoseCrossAccountRole
```

由于创建了流日志，您可以从为传输流配置的目标获取流日志数据。

使用 Amazon Athena 查询流日志

Amazon Athena 是一种交互式查询服务，让您能够使用标准 SQL 分析 Amazon S3 中的数据，例如流日志。您可以将 Athena 与 VPC 流日志结合使用，以快速获得有关流经 VPC 的流量的切实可行的见解。例如，您可以识别虚拟私有云 (VPC) 中的哪些资源是最大流量生成者，或识别拒绝的 TCP 连接最多的 IP 地址。

选项

- 您可以通过生成 CloudFormation 模板来简化和自动完成 VPC 流日志与 Athena 的集成，该模板可创建所需的AWS资源和预定义查询，您可以运行它们以获取有关流经 VPC 的流量的见解。
- 您可以使用 Athena 创建自己的查询。有关更多信息，请参阅 Amazon Athena 用户指南中的使用 Amazon Athena 查询流日志。

定价

运行查询时，您需要承担标准 Amazon Athena 收费。对于按循环计划加载新分区（当您指定分区加载频率但未指定开始和结束日期时）的 Lambda 函数，您需要承担标准 AWS Lambda 收费。

使用预定义的查询

- 使用控制台生成 CloudFormation 模板
- 使用 AWS CLI 生成 CloudFormation 模板
- 运行预定义查询

使用控制台生成 CloudFormation 模板

在将第一个流日志传送到 S3 存储桶后，您可以通过生成 CloudFormation 模板并使用该模板创建堆栈，与 Athena 进行集成。

要求

- 选择的区域必须支持 AWS Lambda 和 Amazon Athena。
- Amazon S3 存储桶必须位于所选区域中。
- 流日志的日志记录格式必须包含您要运行的特定预定义查询所使用的字段。

使用控制台生成模板

1. 请执行下列操作之一：
   - 打开 Amazon VPC 控制台。在导航窗格中，选择 Your VPCs（您的 VPC），然后选择自己的 VPC。
   - 打开 Amazon VPC 控制台。在导航窗格中，选择 Subnets（子网），然后选择您的子网。
   - 打开 Amazon EC2 控制台。在导航窗格中，选择 Network Interfaces（网络接口），然后选择您的网络接口。

2. 在 Flow logs（流日志）选项卡上，选择发布到 Amazon S3 的流日志，然后依次选择 Actions（操作）、Generate Athena integration（生成 Athena 集成）。
3. 指定分区加载频率。如果选择 None（无），则必须使用已过去的日期指定分区的开始日期和结束日期。如果选择Daily（每日）、Weekly（每周）或 Monthly（每月），则分区的开始日期和结束日期是可选设置。如果不指定开始日期和结束日期，CloudFormation 模板创建的 Lambda 函数可以按循环计划加载新分区。
4. 为生成的模板选择或创建 S3 存储桶，为查询结果选择或创建 S3 存储桶。
5. 选择 Generate Athena integration（生成 Athena 集成）。
6. （可选）在成功消息中，选择链接以导航到您为 CloudFormation 模板指定的存储桶，然后自定义该模板。
7. 在成功消息中，选择 Create CloudFormation stack（创建 CloudFormation 堆栈），以在 AWS CloudFormation 控制台中打开 Create Stack (创建堆栈) 向导。生成的 CloudFormation 模板的 URL 在 Template（模板）部分中指定。完成向导以创建模板中指定的资源。

CloudFormation 模板创建的资源

- Athena 数据库。该数据库名称为 vpcflowlogsathenadatabase<flow-logs-subscription-id>。
- Athena 工作组。工作组名称为 <flow-log-subscription-id><partition-load-frequency><start-date><end-date>workgroup
- 与您的流日志记录对应的已分区 Athena 表。表名称为 <flow-log-subscription-id><partition-load-frequency><start-date><end-date>。
- 一组 Athena 命名的查询。有关更多信息，请参阅 预定义查询。
- 一个 Lambda 函数，可按指定的时间表（每日、每周或每月）将新分区加载到表中。
- 授予运行 Lambda 函数的权限的 IAM 角色。

使用 AWS CLI 生成 CloudFormation 模板

在将第一个流日志传送到 S3 存储桶后，您可以生成并使用 CloudFormation 模板与 Athena 集成。

使用下面的 get-flow-logs-integration-template 命令生成 CloudFormation 模板。

```
aws ec2 get-flow-logs-integration-template --cli-input-json file://config.json
```

以下是 `config.json` 文件的示例。

```
{
    "FlowLogId": "fl-12345678901234567",
    "ConfigDeliveryS3DestinationArn": "arn:aws:s3:::my-flow-logs-athena-integration/
templates/",
    "IntegrateServices": {
        "AthenaIntegrations": [
            {
                "IntegrationResultS3DestinationArn": "arn:aws:s3:::my-flow-logs-
analysis/athena-query-results/",
                "PartitionLoadFrequency": "monthly",
                "PartitionStartDate": "2021-01-01T00:00:00",
                "PartitionEndDate": "2021-12-31T00:00:00"
            }
        ]
    }
}
```

使用下面的 create-stack 命令通过所生成的 CloudFormation 模板创建堆栈。

```
aws cloudformation create-stack --stack-name my-vpc-flow-logs --template-body file://
my-cloudformation-template.json
```

运行预定义查询

生成的 CloudFormation 模板提供了一组预定义查询，您可以运行这些查询，以快速获取有关AWS网络中的流量的有意义见解。创建堆栈并验证所有资源都创建正确后，您可以运行其中一个预定义查询。

使用控制台运行预定义查询

1. 打开 Athena 控制台。
2. 在导航窗格中，选择 Query editor（查询编辑器）。在 Workgroups（工作组）下，选择由 CloudFormation 模板创建的工作组。
3. 选择 Saved queries（保存的查询）、选择一个查询、根据需要修改参数，然后运行查询。有关可用预定义查询的列表，请参阅预定义查询。
4. 在 Query results（查询结果）下，查看查询结果。

预定义查询

以下是 Athena 命名查询的完整列表。生成模板时提供的预定义查询取决于流日志的日志记录格式包含的字段。因此，模板可能不会包含所有这些预定义查询。

- VpcFlowLogsAcceptedTraffic — 根据您的安全组和网络 ACL 允许的 TCP 连接。
- VpcFlowLogsAdminPortTraffic – 流量最多的前 10 个 IP 地址，以通过管理端口服务请求的应用程序的记录为准。
- VpcFlowLogsIPv4Traffic — 记录的 IPv4 流量的总字节数。
- VpcFlowLogsIPv6Traffic — 记录的 IPv6 流量的总字节数。
- VpcFlowLogsRejectedTCPTraffic — 根据您的安全组或网络 ACL 拒绝的 TCP 连接。
- VpcFlowLogsRejectedTraffic — 根据您的安全组或网络 ACL 拒绝的流量。
- VpcFlowLogsSshRdpTraffic — SSH 和 RDP 流量。
- VpcFlowLogsTopTalkers — 记录的流量最多的 50 个 IP 地址。
- VpcFlowLogsTopTalkersPacketLevel — 记录的流量最多的 50 个数据包级别 IP 地址。
- VpcFlowLogsTopTalkingInstances — 记录的流量最多的 50 个实例的 ID。
- VpcFlowLogsTopTalkingSubnets — 记录的流量最多的 50 个子网的 ID。
- VpcFlowLogsTopTCPTraffic — 为源 IP 地址记录的所有 TCP 流量。
- VpcFlowLogsTotalBytesTransferred — 记录的字节数最多的 50 对源和目标 IP 地址。
- VpcFlowLogsTotalBytesTransferredPacketLevel — 记录的字节数最多的 50 对数据包级别的源和目标 IP 地址。
- VpcFlowLogsTrafficFrmSrcAddr — 针对特定源 IP 地址记录的流量。
- VpcFlowLogsTrafficToDstAddr — 针对特定目标 IP 地址记录的流量。

VPC 流日志疑难解答

下面是您在使用流日志时可能遇到的问题。

问题

- 未完成的流日志记录
- 流日志处于活动状态，但没有流日志记录或日志组
- “LogDestinationNotFoundException”或“Access Denied for LogDestination”错误
- 超出 Amazon S3 存储桶策略限制
- LogDestination 无法送达
- 流日志数据大小与账单数据不匹配

未完成的流日志记录

问题

您的流日志记录不完整或者已不再发布。

原因

将流日志传送到 CloudWatch Logs 日志组时可能会出现问题，或者可能存在 SkipData 条目。

解决方案

查看 VPC、子网或网络接口的流日志选项卡。请注意，您无法描述与您共享的 VPC 或子网的流日志，但可以描述在与您共享的 VPC 或子网中由您创建的网络接口的流日志。如果存在任何错误，将会在状态列中显示。或者，使用 describe-flow-logs 命令，然后检查 `DeliverLogsErrorMessage` 字段中返回的值。

状态可能出现的错误值如下所示：

- `Rate limited`：如果应用了 CloudWatch Logs 日志节流，则当某个网络接口的流日志记录数超过了可以在特定时间范围内发布的最大记录数时，会出现此错误。如果已达到您可创建的 CloudWatch Logs 日志组数量的配额，也可能出现此错误。有关更多信息，请参阅《Amazon CloudWatch 用户指南》中的 CloudWatch 服务配额。
- `Access error`：出现此错误的原因为以下之一：
  - 您的流日志的 IAM 角色没有足够的权限，无法将流日志记录发布到 CloudWatch 日志组
  - IAM 角色与流日志服务没有信任关系
  - 信任关系不指定用作委托人的流日志服务

  有关更多信息，请参阅 用于将流日志发布到 CloudWatch Logs 的 IAM 角色。
- `Unknown error`：流日志服务中出现内部错误。

流日志处于活动状态，但没有流日志记录或日志组

问题

您已创建流日志，并且 Amazon VPC 或 Amazon EC2 控制台会将流日志显示为 `Active`。但是，您无法看到 CloudWatch Logs 中的任何日志流或 Amazon S3 存储桶中的日志文件。

可能的原因

- 流日志仍在创建中。在某些情况下，当您为要创建的日志组创建流日志后，有时会需要数十分钟或更长时间才会显示数据。
- 还没有为您的网络接口记录任何流量。只有在记录流量时，才会创建 CloudWatch Logs 中的日志组。

解决方案

等待几分钟，以便系统创建日志组，或者记录流量。

“LogDestinationNotFoundException”或“Access Denied for LogDestination”错误

问题

当您创建流日志时，会出现 `Access Denied for LogDestination` 或 `LogDestinationNotFoundException` 错误。

可能的原因

- 创建将数据发布到 Amazon S3 存储桶的流日志时，此错误表示找不到指定的 S3 存储桶或存储桶策略不允许将日志传送到存储桶。
- 创建将数据发布到 Amazon CloudWatch Logs 的流日志时，此错误表示 IAM 角色不允许将日志传送到日志组。

解决方案

- 发布到 Amazon S3 时，请确保您指定了现有 S3 存储桶的 ARN，并且该 ARN 的格式正确。如果您不拥有 S3 存储桶，请验证存储桶策略具有所需的权限，并在 ARN 中使用正确的账户 ID 和存储桶名称。
- 发布到 CloudWatch Logs 时，请验证 IAM 角色具有所需的权限。

超出 Amazon S3 存储桶策略限制

问题

在您尝试创建流日志时收到了以下错误：`LogDestinationPermissionIssueException`

可能的原因

Amazon S3 存储桶策略有 20 KB 的大小限制。

每次创建发布到 Amazon S3 存储桶的流日志时，我们都会自动将指定的存储桶 ARN（包括文件夹路径）添加到存储桶策略中的 `Resource` 元素。

创建发布到同一个存储桶的多个流日志可能会导致超出存储桶策略限制。

解决方案

- 通过删除不再需要的流日志条目来清理存储桶策略。
- 通过使用以下内容替换各个流日志条目，为整个存储桶授予权限。

```
arn:aws:s3:::bucket_name/*
```

如果您授予整个存储桶的权限，则新的流日志订阅不会向存储桶策略添加新权限。

LogDestination 无法送达

问题

在您尝试创建流日志时收到了以下错误：`LogDestination <bucket name> is undeliverable`

可能的原因

目标 Amazon S3 存储桶将使用 AWS KMS（SSE-KMS）和服务器端加密进行加密，并且存储桶的默认加密方式为 KMS 密钥 ID。

解决方案

该值必须为 KMS 密钥 ARN。将默认 S3 加密类型从 KMS 密钥 ID 更改为 KMS 密钥 ARN。有关更多信息，请参阅《Amazon Simple Storage Service 控制台用户指南》中的 配置默认加密。

流日志数据大小与账单数据不匹配

问题

流日志的总数据大小与账单数据报告的大小不匹配。

可能的原因

您的流日志中可能有 SKIPDATA 条目。有关 SKIPDATA 条目的说明，请参阅 无数据和跳过的记录。

解决方案

通过在日志状态字段中查询日志中的不同条目，确认日志条目中是否存在 SKIPDATA 条目。

检查 SKIPDATA 的示例查询：

CW Insights：

```
fields @timestamp, @message, @logStream, @log
| filter interfaceId = 'eni-123'
| stats count(*) by interfaceId, logStatus
| sort by interfaceId, logStatus
```

Athena：

```
SELECT log_status, interface_id, count(1)
FROM vpc_flow_logs
WHERE interface_id IN ('eni-1', 'eni-2', 'eni-3')
GROUP BY log_status, interface_id
```

VPC 的 CloudWatch 指标

Amazon VPC 会将有关 VPC 的数据发布到 Amazon CloudWatch。您可以将有关 VPC 的统计数据作为一组有序时间序列数据（称作指标）进行检索。可将指标视为要监控的变量，而将数据视为该变量随时间变化的值。有关更多信息，请参阅 Amazon CloudWatch 用户指南。

内容

- NAU 指标与维度
- 启用或禁用 NAU 监控
- NAU CloudWatch 告警示例

NAU 指标与维度

网络地址用量 (NAU) 是应用于虚拟网络中资源的指标，可以帮助您规划和监控 VPC 的大小。监控 NAU 不收取任何费用。监控 NAU 非常有用，因为如果耗尽了 VPC 的 NAU 限额或对等 NAU 限额，您将无法启动新的 EC2 实例或预调配新的资源（例如网络负载均衡器、VPC 端点、Lambda 函数、中转网关连接或 NAT 网关）。

如果您为 VPC 启用了网络地址用量监控，Amazon VPC 会将与 NAU 相关的指标发送到 Amazon CloudWatch。VPC 的大小取决于 VPC 包含的网络地址用量（NAU）单元数。

您可以使用这些指标来了解您的 VPC 增长率、预测 VPC 何时达到其大小限制，或者在超过其大小阈值时创建告警。

AWS/EC2 命名空间包括以下用于监控 NAU 的指标。

指标	描述
NetworkAddressUsage	每个 VPC 的 NAU 计数。 报告标准 • 每 24 小时。 尺寸 • 名称：Per-VPC Metrics，值：VPC ID。
NetworkAddressUsagePeered	VPC 和与之对等的所有 VPC 的 NAU 计数。 报告标准 • 每 24 小时。 尺寸 • 名称：Per-VPC Metrics，值：VPC ID。

AWS/Usage 命名空间包括以下用于监控 NAU 的指标。

指标	描述
ResourceCount	每个 VPC 的 NAU 计数。 报告标准 • 每 24 小时。 尺寸

指标	描述
	• 名称：Service，值：EC2 • 名称：Type，值：Resource • 名称：Resource，值：VPC ID。 • 名称：Class，值：NetworkAddressUsage
ResourceCount	VPC 和与之对等的所有 VPC 的 NAU 计数。 报告标准 • 每 24 小时。 尺寸 • 名称：Service，值：EC2 • 名称：Type，值：Resource • 名称：Resource，值：VPC ID。 • 名称：Class，值：NetworkAddressUsagePeered
ResourceCount	跨 VPC 的 NAU 用量组合视图。 报告标准 • 每 24 小时。 尺寸 • 名称：Service，值：EC2 • 名称：Type，值：Resource • 名称：Resource，值：VPC • 名称：Class，值：NetworkAddressUsage

指标	描述
ResourceCount	跨对等 VPC 的 NAU 用量组合视图。 报告标准 • 每 24 小时。 尺寸 • 名称：Service，值：EC2 • 名称：Type，值：Resource • 名称：Resource，值：VPC • 名称：Class，值：NetworkAddressUsagePeered

启用或禁用 NAU 监控

要在 CloudWatch 中查看 NAU 指标，必须先在每个要监控的 VPC 上启用监控。

启用或禁用监控 NAU

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Your VPCs(您的 VPC)。
3. 选中 VPC 的复选框。
4. 依次选择 Actions（操作）、Edit VPC settings（编辑 VPC 设置）。
5. 请执行以下操作之一：

   - 要启用监控，依次选择 Network mapping units metrics settings（网络映射单元指标设置），Enable network address usage metrics（启用网络地址用量指标）。
   - 要禁用监控，请清除 Network mapping units metrics settings（网络映射单元指标设置）和 Enable network address usage metrics（启用网络地址用量指标）。

使用命令行启用或禁用监控

- modify-vpc-attribute (AWS CLI)

- Edit-EC2VpcAttribute (AWS Tools for Windows PowerShell)

NAU CloudWatch 告警示例

您可以使用以下 AWS CLI 命令和示例 `.json` 创建 Amazon CloudWatch 警报和 SNS 通知，以 50,000 个 NAU 为阈值跟踪 VPC 的 NAU 利用率。此示例要求您先创建 Amazon SNS 主题。有关更多信息，请参阅 Amazon Simple Notification Service 开发人员指南中的 Amazon SNS 入门。

```
aws cloudwatch put-metric-alarm --cli-input-json file://nau-alarm.json
```

以下是 `nau-alarm.json` 的示例。

```
{
    "Namespace": "AWS/EC2",
    "MetricName": "NetworkAddressUsage",
    "Dimensions": [{
        "Name": "Per-VPC Metrics",
        "Value": "vpc-0123456798"
    }],
    "AlarmActions": ["arn:aws:sns:us-west-1:123456789012:my_sns_topic"],
    "ComparisonOperator": "GreaterThanThreshold",
    "Period": 86400,
    "EvaluationPeriods": 1,
    "Threshold": 50000,
    "AlarmDescription": "Tracks NAU utilization of the VPC with 50k NAUs as the
 threshold",
    "AlarmName": "VPC NAU Utilization",
    "Statistic": "Maximum"
}
```

了解账单和使用情况报告中的 Amazon VPC 代码

当您使用 Amazon VPC 时，我们会在 AWS 账单和使用情况报告中包含相关代码。查看这些代码有助于您了解 Amazon VPC 的成本和使用模式。跟踪和管理支出对于优化成本至关重要。

下表列出了账单和使用情况报告中出现的 Amazon VPC 代码。有关账单和使用情况报告中使用的区域代码列表，请参阅 AWS Region billing codes。

以下各项的账单代码：

- IP 地址管理

- VPC 端点
- 中转网关
- 网络分析
- 流量镜像
- VPC Lattice
- 跨账户/跨区域资源

相关资源

- Amazon VPC 定价
- AWS PrivateLink 定价
- AWS Transit Gateway 定价
- Amazon VPC Lattice 定价

IP 地址管理

代码	描述	单位	粒度
region-PublicIPv4:InUseAddress	某个资源使用公有 IPv4 地址的时间。	Hours	每秒
region-PublicIPv4:IdleAddress	某个资源未使用公有 IPv4 地址的时间。	Hours	每秒
region-PublicIPv4:ContiguousBlock	使用了 Amazon 提供的连续 IPv4 块中的公有 IPv4 地址。	Hours	每小时
region-IPAddres	IPAM 高级套餐管理 IP 地址的时间。	Hours	每小时

代码	描述	单位	粒度
sManager-IP-Hours			

VPC 端点

代码	描述	单位	粒度
region-VpcEndpoint-Hours	预置接口 VPC 端点的时间。	Hours	每小时
region-VpcEndpoint-Bytes	接口 VPC 端点处理的数据。	GB	每小时
region-VpcEndpoint-GWLBE-Hours	预置网关负载均衡器端点的时间。	Hours	每小时
region-VpcEndpoint-GWLBE-Bytes	网关负载均衡器端点处理的数据。	GB	每小时

中转网关

代码	描述	单位	粒度
region-TransitG	使用了中转网关连接。	Hours	每小时

代码	描述	单位	粒度
ateway-Hours			
region-TransitGateway-Bytes	中转网关处理的数据。	GB	每小时
region-TGW-Multicast-Consumer-Bytes	组播接收器实例处理的数据。	GB	每小时

网络分析

代码	描述	单位	粒度
region-Analysis-Runs	Reachability Analyzer 分析的网络路径数量。	计数	每次分析
region-NetworkInterface-Assessment	网络访问分析器分析的网络接口数量。	计数	每次评测

流量镜像

代码	描述	单位	粒度
region-ENI-Mirror	为流量镜像配置网络接口的时间。	Hours	每小时

VPC Lattice

代码	描述	单位	粒度
region-VPCLattice-Service-Hourly	VPC Lattice 服务的运行时间。	Hours	每小时
region-VPCLattice-DataProcessing-Bytes	VPC Lattice 服务处理的数据。	GB	每小时
region-VPCLattice-RequestCount-Free	免费 HTTP 请求和 TCP 连接。	计数	每小时
region-VpcLattice-Service-Network-Resource-Hours	VPC Lattice 服务网络的运行时间。	Hours	每小时

跨账户/跨区域资源

代码	描述	单位	粒度
region-VpcResou	从提供器资源跨账户或跨区域传输的数据。	GB	每小时

代码	描述	单位	粒度
rce-Provider-Bytes			
region-VpcResource-Consumer-Bytes	消费器资源跨账户或跨区域传输的数据。	GB	每小时

管理 Amazon Virtual Private Cloud 的安全责任

AWS 十分重视云安全性。作为 AWS 客户，您将从专为满足大多数安全敏感型企业的要求而打造的数据中心和网络架构中受益。

安全性是 AWS 和您的共同责任。责任共担模式将其描述为云的安全性和云中的安全性：

- 云安全性 – AWS 负责保护在 AWS 云中运行 AWS 服务的基础结构。AWS 还向您提供可安全使用的服务。第三方审核员定期测试和验证我们的安全性的有效性，作为 AWS Compliance Programs 的一部分。要了解适用于 Amazon Virtual Private Cloud 的合规性计划，请参阅合规性计划范围内的 AWS 服务。
- 云中的安全性：您的责任由您使用的 AWS 服务决定。您还需要对其他因素负责，包括您的数据的敏感性、您公司的要求以及适用的法律法规。

此文档将帮助您了解如何在使用 Amazon VPC 时应用责任共担模式。以下主题说明如何配置 Amazon VPC 以实现您的安全性和合规性目标。您还会了解如何使用其他 AWS 服务以帮助您监控和保护 Amazon VPC 资源。

内容

- 确保 Amazon Virtual Private Cloud 中的数据保护
- 适用于 Amazon VPC 的 Identity and Access Management
- Amazon VPC 中的基础设施安全性
- 使用安全组控制指向 AWS 资源的流量
- 使用网络访问控制列表控制子网流量
- Amazon Virtual Private Cloud 中的恢复能力
- Amazon Virtual Private Cloud 的合规性验证
- 屏蔽 VPC 和子网的公共访问权限
- VPC 的安全最佳实践

确保 Amazon Virtual Private Cloud 中的数据保护

AWS 责任共担模型适用于 Amazon Virtual Private Cloud 中的数据保护。如该模式中所述，AWS 负责保护运行所有 AWS Cloud 的全球基础结构。您负责维护对托管在此基础结构上的内容的控制。您还负责您所使用的 AWS 服务 的安全配置和管理任务。有关数据隐私的更多信息，请参阅数据隐私常见

问题。有关欧洲数据保护的信息，请参阅 AWS Security Blog 上的 AWS Shared Responsibility Model and GDPR 博客文章。

出于数据保护目的，建议您保护 AWS 账户 凭证并使用 AWS IAM Identity Center 或 AWS Identity and Access Management（IAM）设置单个用户。这样，每个用户只获得履行其工作职责所需的权限。还建议您通过以下方式保护数据：

- 对每个账户使用多重身份验证（MFA）。
- 使用 SSL/TLS 与 AWS 资源进行通信。我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 使用 AWS CloudTrail 设置 API 和用户活动日记账记录。有关使用 CloudTrail 跟踪来捕获 AWS 活动的信息，请参阅《AWS CloudTrail 用户指南》中的使用 CloudTrail 跟踪。
- 使用 AWS 加密解决方案以及 AWS 服务中的所有默认安全控制。
- 使用高级托管安全服务（例如 Amazon Macie），它有助于发现和保护存储在 Amazon S3 中的敏感数据。
- 如果在通过命令行界面或 API 访问 AWS 时需要经过 FIPS 140-3 验证的加密模块，请使用 FIPS 端点。有关可用的 FIPS 端点的更多信息，请参阅《美国联邦信息处理标准（FIPS）第 140-3 版》。

强烈建议您切勿将机密信息或敏感信息（如您客户的电子邮件地址）放入标签或自由格式文本字段（如名称字段）。这包括使用控制台、API、AWS CLI 或 AWS SDK 处理 Amazon VPC 或其他 AWS 服务 时。在用于名称的标签或自由格式文本字段中输入的任何数据都可能会用于计费或诊断日志。如果您向外部服务器提供网址，强烈建议您不要在网址中包含凭证信息来验证对该服务器的请求。

确保 Amazon VPC 中的互联网络流量隐私

Amazon Virtual Private Cloud 提供三种功能，以供您用来提高和监控 Virtual Private Cloud (VPC) 的安全性：

- 安全组：安全组在资源级别（例如 EC2 实例）允许特定入站和出站流量。启动实例时，您可将其与一个或多个安全组关联。在您的 VPC 中的每项实例都可能属于不同的安全组集合。如果您在启动实例时没有指定安全组，则实例会自动与其 VPC 的默认安全组关联。有关更多信息，请参阅 安全组。
- 网络访问控制列表 (ACL)：网络 ACL 在子网级别允许或拒绝特定的入站和出站流量。有关更多信息，请参阅 使用网络访问控制列表控制子网流量。
- 流日志：流日志捕获有关在您的 VPC 中传入和传出网络接口的 IP 流量的信息。您可以为 VPC、子网或各个网络接口创建流日志。流日志数据将发布到 CloudWatch Logs 或 Amazon S3，可帮助您诊断过于严格或过于宽松的安全组和网络 ACL 规则。有关更多信息，请参阅 使用 VPC 流日志记录 IP 流量。

- 流量镜像：您可以从 Amazon EC2 实例的弹性网络接口复制网络流量。然后将流量发送到带外安全和监控设备。有关更多信息，请参阅流量镜像指南。

适用于 Amazon VPC 的 Identity and Access Management

AWS Identity and Access Management（IAM）是一项 AWS 服务，可以帮助管理员安全地控制对 AWS 资源的访问。IAM 管理员控制谁可以通过身份验证（登录）和授权（具有权限）使用 Amazon VPC 资源。IAM 是一项无需额外费用即可使用的AWS 服务。

内容

- 受众
- 使用身份进行身份验证
- 使用策略管理访问
- Amazon VPC 如何与 IAM 配合使用
- Amazon VPC 策略示例
- Amazon VPC 身份和访问疑难解答
- 适用于 Amazon Virtual Private Cloud 的 AWS 托管策略

受众

如何使用 AWS Identity and Access Management (IAM) 因您在 Amazon VPC 中执行的操作而异。

服务用户 – 如果您使用 Amazon VPC 服务来完成工作，则管理员会为您提供所需的凭证和权限。随着您使用更多 Amazon VPC 功能来完成工作，您可能需要额外权限。了解如何管理访问权限可帮助您向管理员请求适合的权限。如果您无法访问 Amazon VPC 中的功能，请参阅Amazon VPC 身份和访问疑难解答。

服务管理员 – 如果您在公司负责管理 Amazon VPC 资源，您可能对 Amazon VPC 具有完全访问权限。您有责任确定您的员工应访问哪些 Amazon VPC 功能和资源。您向 IAM 管理员提交请求以更改服务用户的权限。请查看该页面上的信息以了解 IAM 的基本概念。要了解有关您的公司如何将 IAM 与 Amazon VPC 搭配使用的更多信息，请参阅Amazon VPC 如何与 IAM 配合使用。

IAM 管理员 – 如果您是 IAM 管理员，您可能希望了解如何编写策略以管理对 Amazon VPC 的访问的详细信息。要查看示例策略，请参阅 Amazon VPC 策略示例。

使用身份进行身份验证

身份验证是您使用身份凭证登录 AWS 的方法。您必须作为 AWS 账户根用户、IAM 用户或通过代入 IAM 角色进行身份验证（登录到 AWS）。

您可以使用通过身份源提供的凭证以联合身份登录到 AWS。AWS IAM Identity Center（IAM Identity Center）用户、您公司的单点登录身份验证以及您的 Google 或 Facebook 凭证都是联合身份的示例。当您以联合身份登录时，您的管理员以前使用 IAM 角色设置了身份联合验证。当您使用联合身份验证访问 AWS 时，您就是在间接代入角色。

根据您的用户类型，您可以登录 AWS Management Console 或 AWS 访问门户。有关登录到 AWS 的更多信息，请参阅《AWS 登录 User Guide》中的 How to sign in to your AWS 账户。

如果您以编程方式访问 AWS，则 AWS 将提供软件开发工具包（SDK）和命令行界面（CLI），以便使用您的凭证以加密方式签署您的请求。如果您不使用 AWS 工具，则必须自行对请求签名。有关使用推荐的方法自行签署请求的更多信息，请参阅《IAM 用户指南》中的用于签署 API 请求的 AWS 签名版本 4。

无论使用何种身份验证方法，您都可能需要提供其他安全信息。例如，AWS 建议您使用多重身份验证（MFA）来提高账户的安全性。要了解更多信息，请参阅《AWS IAM Identity Center 用户指南》中的多重身份验证和《IAM 用户指南》中的 IAM 中的 AWS 多重身份验证。

AWS 账户根用户

当您创建 AWS 账户时，最初使用的是一个对账户中所有 AWS 服务和资源拥有完全访问权限的登录身份。此身份称为 AWS 账户根用户，使用您创建账户时所用的电子邮件地址和密码登录，即可获得该身份。强烈建议您不要使用根用户执行日常任务。保护好根用户凭证，并使用这些凭证来执行仅根用户可以执行的任务。有关需要您以根用户身份登录的任务的完整列表，请参阅《IAM 用户指南》中的需要根用户凭证的任务。

IAM 用户和群组

IAM 用户是 AWS 账户内对某个人员或应用程序具有特定权限的一个身份。在可能的情况下，我们建议使用临时凭证，而不是创建具有长期凭证（如密码和访问密钥）的 IAM 用户。但是，如果您有一些特定的使用场景需要长期凭证以及 IAM 用户，建议您轮换访问密钥。有关更多信息，请参阅《IAM 用户指南》中的对于需要长期凭证的用例，应在需要时更新访问密钥。

IAM 组是一个指定一组 IAM 用户的身份。您不能使用组的身份登录。您可以使用组来一次性为多个用户指定权限。如果有大量用户，使用组可以更轻松地管理用户权限。例如，您可能具有一个名为 IAMAdmins 的组，并为该组授予权限以管理 IAM 资源。

用户与角色不同。用户唯一地与某个人员或应用程序关联，而角色旨在让需要它的任何人代入。用户具有永久的长期凭证，而角色提供临时凭证。要了解更多信息，请参阅《IAM 用户指南》中的 IAM 用户的使用案例。

IAM 角色

IAM 角色是 AWS 账户中具有特定权限的身份。它类似于 IAM 用户，但与特定人员不关联。要在 AWS Management Console中临时代入 IAM 角色，可以从用户切换到 IAM 角色（控制台）。您可以调用 AWS CLI 或 AWS API 操作或使用自定义网址以代入角色。有关使用角色的方法的更多信息，请参阅《IAM 用户指南》中的代入角色的方法。

具有临时凭证的 IAM 角色在以下情况下很有用：

- 联合用户访问：要向联合身份分配权限，请创建角色并为角色定义权限。当联合身份进行身份验证时，该身份将与角色相关联并被授予由此角色定义的权限。有关用于联合身份验证的角色的信息，请参阅《IAM 用户指南》中的针对第三方身份提供商创建角色（联合身份验证）。如果您使用 IAM Identity Center，则需要配置权限集。为控制您的身份在进行身份验证后可以访问的内容，IAM Identity Center 将权限集与 IAM 中的角色相关联。有关权限集的信息，请参阅《AWS IAM Identity Center 用户指南》中的权限集。
- 临时 IAM 用户权限：IAM 用户可代入 IAM 用户或角色，以暂时获得针对特定任务的不同权限。
- 跨账户存取：您可以使用 IAM 角色以允许不同账户中的某个人（可信主体）访问您的账户中的资源。角色是授予跨账户访问权限的主要方式。但是，对于某些 AWS 服务，您可以将策略直接附加到资源（而不是使用角色作为代理）。要了解用于跨账户访问的角色和基于资源的策略之间的差别，请参阅 IAM 用户指南中的 IAM 中的跨账户资源访问。
- 跨服务访问：某些 AWS 服务使用其他 AWS 服务中的特征。例如，当您在某个服务中进行调用时，该服务通常会在 Amazon EC2 中运行应用程序或在 Simple Storage Service（Amazon S3）中存储对象。服务可能会使用发出调用的主体的权限、使用服务角色或使用服务相关角色来执行此操作。
  - 转发访问会话（FAS）：当您使用 IAM 用户或角色在 AWS 中执行操作时，您将被视为主体。使用某些服务时，您可能会执行一个操作，然后此操作在其他服务中启动另一个操作。FAS 使用主体调用 AWS 服务的权限，结合请求的 AWS 服务，向下游服务发出请求。只有在服务收到需要与其他 AWS 服务或资源交互才能完成的请求时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两项操作的权限。有关发出 FAS 请求时的策略详情，请参阅转发访问会话。
  - 服务角色 - 服务角色是服务代表您在您的账户中执行操作而分派的 IAM 角色。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的创建向 AWS 服务委派权限的角色。

- 服务相关角色：服务相关角色是与 AWS 服务 关联的一种服务角色。服务可以代入代表您执行操作的角色。服务相关角色显示在您的 AWS 账户 中，并由该服务拥有。IAM 管理员可以查看但不能编辑服务相关角色的权限。
- 在 Amazon EC2 上运行的应用程序：您可以使用 IAM 角色管理在 EC2 实例上运行并发出 AWS CLI 或 AWS API 请求的应用程序的临时凭证。这优先于在 EC2 实例中存储访问密钥。要将 AWS 角色分配给 EC2 实例并使其对该实例的所有应用程序可用，您可以创建一个附加到实例的实例配置文件。实例配置文件包含角色，并使 EC2 实例上运行的程序能够获得临时凭证。有关更多信息，请参阅《IAM 用户指南》中的使用 IAM 角色向在 Amazon EC2 实例上运行的应用程序授予权限。

使用策略管理访问

您将创建策略并将其附加到 AWS 身份或资源，以控制 AWS 中的访问。策略是 AWS 中的对象；在与身份或资源相关联时，策略定义它们的权限。在主体（用户、根用户或角色会话）发出请求时，AWS 将评估这些策略。策略中的权限确定是允许还是拒绝请求。大多数策略在 AWS 中存储为 JSON 文档。有关 JSON 策略文档的结构和内容的更多信息，请参阅 IAM 用户指南中的 JSON 策略概览。

管理员可以使用 AWS JSON 策略来指定谁有权访问什么内容。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

默认情况下，用户和角色没有权限。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM 策略。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

IAM 策略定义操作的权限，无关乎您使用哪种方法执行操作。例如，假设您有一个允许 `iam:GetRole` 操作的策略。具有该策略的用户可以从 AWS Management Console、AWS CLI 或 AWS API 获取角色信息。

基于身份的策略

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的使用客户托管策略定义自定义 IAM 权限。

基于身份的策略可以进一步归类为内联策略或托管式策略。内联策略直接嵌入单个用户、组或角色中。托管策略是可以附加到 AWS 账户 中的多个用户、组和角色的独立策略。托管式策略包括 AWS 托管式策略和客户托管策略。要了解如何在托管策略和内联策略之间进行选择，请参阅《IAM 用户指南》中的在托管策略与内联策略之间进行选择。

基于资源的策略

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中指定主体。主体可以包括账户、用户、角色、联合用户或 AWS 服务。

基于资源的策略是位于该服务中的内联策略。您不能在基于资源的策略中使用来自 IAM 的 AWS 托管式策略。

访问控制列表（ACL）

访问控制列表（ACL）控制哪些主体（账户成员、用户或角色）有权访问资源。ACL 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

Amazon S3、AWS WAF 和 Amazon VPC 是支持 ACL 的服务示例。要了解有关 ACL 的更多信息，请参阅《Amazon Simple Storage Service 开发人员指南》中的访问控制列表（ACL）概览。

其他策略类型

AWS 支持额外的、不太常用的策略类型。这些策略类型可以设置更常用的策略类型向您授予的最大权限。

- 权限边界：权限边界是一个高级特征，用于设置基于身份的策略可以为 IAM 实体（IAM 用户或角色）授予的最大权限。您可为实体设置权限边界。这些结果权限是实体基于身份的策略及其权限边界的交集。在 `Principal` 中指定用户或角色的基于资源的策略不受权限边界限制。任一项策略中的显式拒绝将覆盖允许。有关权限边界的更多信息，请参阅IAM 用户指南中的 IAM 实体的权限边界。
- 服务控制策略（SCP）– SCP 是 JSON 策略，指定了组织或组织单元（OU）在 AWS Organizations 中的最大权限。AWS Organizations 服务可以分组和集中管理您的企业拥有的多个 AWS 账户 账户。如果在组织内启用了所有特征，则可对任意或全部账户应用服务控制策略（SCP）。SCP 限制成员账户中实体（包括每个 AWS 账户根用户）的权限。有关组织和 SCP 的更多信息，请参阅《AWS Organizations 用户指南》中的服务控制策略。
- 资源控制策略（RCP）– RCP 是 JSON 策略，您可以使用它们设置账户中资源的最大可用权限，而无需更新附加到您拥有的每个资源的 IAM 策略。RCP 限制了成员账户中资源的权限，并可能影响身份（包括 AWS 账户根用户）的有效权限，无论这些身份是否属于您的组织。有关 Organizations 和 RCP（包括支持 RCP 的 AWS 服务列表）的更多信息，请参阅《AWS Organizations User Guide》中的 Resource control policies (RCPs)。

- 会话策略：会话策略是当您以编程方式为角色或联合用户创建临时会话时作为参数传递的高级策略。结果会话的权限是用户或角色的基于身份的策略和会话策略的交集。权限也可以来自基于资源的策略。任一项策略中的显式拒绝将覆盖允许。有关更多信息，请参阅 IAM 用户指南中的会话策略。

多个策略类型

当多个类型的策略应用于一个请求时，生成的权限更加复杂和难以理解。要了解如何 AWS 确定在涉及多种策略类型时是否允许请求，请参阅《IAM 用户指南》中的策略评估逻辑。

Amazon VPC 如何与 IAM 配合使用

在使用 IAM 管理对 Amazon VPC 的访问权限之前，您应该了解哪些 IAM 功能可用于 Amazon VPC。要大致了解 Amazon VPC 和其他 AWS 服务如何与 IAM 一起使用，请参阅《IAM 用户指南》中的 与 IAM 一起使用的 AWS 服务。

目录

- 操作
- 资源
- 条件键
- 基于 Amazon VPC 资源的策略
- 基于标签的授权
- IAM 角色

使用 IAM 基于身份的策略，您可以指定允许或拒绝的操作。对于部分操作，您可以指定允许或拒绝操作和资源，并指定在什么条件下允许或拒绝操作。Amazon VPC 支持特定的操作、资源和条件键。要了解在 JSON 策略中使用的所有元素，请参阅《IAM 用户指南》中的 IAM JSON 策略元素参考。

操作

管理员可以使用 AWS JSON 策略来指定谁有权访问什么内容。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

JSON 策略的 `Action` 元素描述可用于在策略中允许或拒绝访问的操作。策略操作通常与关联的 AWS API 操作同名。有一些例外情况，例如没有匹配 API 操作的仅限权限 操作。还有一些操作需要在策略中执行多个操作。这些附加操作称为相关操作。

在策略中包含操作以授予执行相关操作的权限。

Amazon VPC 与 Amazon EC2 共享其 API 命名空间。Amazon VPC 中的策略操作在操作前面使用以下前缀：ec2:。例如，要授予用户通过 CreateVpc API 操作创建 VPC 的权限，您需要授予对 ec2:CreateVpc 操作的访问权限。策略语句必须包含 Action 或 NotAction 元素。

要在单个语句中指定多项操作，请使用逗号将它们隔开，如下例所示。

```
"Action": [
      "ec2:action1",
      "ec2:action2"
]
```

您也可以使用通配符 (*) 指定多个操作。例如，要指定以单词 Describe 开头的所有操作，请包括以下操作。

```
"Action": "ec2:Describe*"
```

有关 Amazon EC2 操作的列表，请参阅《服务授权参考》中的 Amazon EC2 定义的操作。

资源

管理员可以使用 AWS JSON 策略来指定谁有权访问什么内容。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

Resource JSON 策略元素指定要向其应用操作的一个或多个对象。语句必须包含 Resource 或 NotResource 元素。作为最佳实践，请使用其 Amazon 资源名称（ARN）指定资源。对于支持特定资源类型（称为资源级权限）的操作，您可以执行此操作。

对于不支持资源级权限的操作（如列出操作），请使用通配符（*）指示语句应用于所有资源。

```
"Resource": "*"
```

VPC 资源具有下例中显示的 ARN。

```
arn:${Partition}:ec2:${Region}:${Account}:vpc/${VpcId}
```

例如，要在语句中指定 vpc-1234567890abcdef0 VPC，请使用以下示例中显示的 ARN。

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:vpc/vpc-1234567890abcdef0"
```

要指定在特定区域中属于特定账户的所有 VPC，请使用通配符 (*)。

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:vpc/*"
```

无法对特定资源执行某些 Amazon VPC 操作，例如，用于创建资源的操作。在这些情况下，您必须使用通配符 (*)。

```
"Resource": "*"
```

许多 Amazon EC2 API 操作涉及多种资源。要在单个语句中指定多个资源，请使用逗号分隔 ARN。

```
"Resource": [
      "resource1",
      "resource2"
]
```

有关 Amazon VPC 资源类型及其 ARN 的列表，请参阅《服务授权参考》中的 Amazon EC2 定义的资源。

条件键

管理员可以使用 AWS JSON 策略来指定谁有权访问什么内容。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

在 Condition 元素（或 Condition 块）中，可以指定语句生效的条件。Condition 元素是可选的。您可以创建使用条件运算符（例如，等于或小于）的条件表达式，以使策略中的条件与请求中的值相匹配。

如果您在一个语句中指定多个 Condition 元素，或在单个 Condition 元素中指定多个键，则 AWS 使用逻辑 AND 运算评估它们。如果您为单个条件键指定多个值，则 AWS 使用逻辑 OR 运算来评估条件。在授予语句的权限之前必须满足所有的条件。

在指定条件时，您也可以使用占位符变量。例如，只有在使用 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。有关更多信息，请参阅《IAM 用户指南》中的 IAM 策略元素：变量和标签。

AWS 支持全局条件键和特定于服务的条件键。要查看所有 AWS 全局条件键，请参阅《IAM 用户指南》中的 AWS 全局条件上下文键。

所有 Amazon EC2 操作都支持 aws:RequestedRegion 和 ec2:Region 条件键。有关更多信息，请参阅 示例：仅允许访问特定区域。

Amazon VPC 定义了自己的一组条件键，还支持使用一些全局条件键。要查看 Amazon VPC 条件键的列表，请参阅《服务授权参考》中的 Amazon EC2 的条件键。要了解您可以对哪些操作和资源使用条件键，请参阅 Amazon EC2 定义的操作。

基于 Amazon VPC 资源的策略

基于资源的策略是 JSON 策略文档，它们指定了指定委托人可在 Amazon VPC 资源上执行的操作以及在什么条件下可执行。

要启用跨账户访问，您可以将整个账户或其他账户中的 IAM 实体指定为基于资源的策略中的委托人。将跨账户委托人添加到基于资源的策略只是建立信任关系工作的一半而已。当委托人和资源位于不同的AWS账户中时，还必须授予委托人实体对资源的访问权限。通过将基于身份的策略附加到实体以授予权限。但是，如果基于资源的策略向同一个账户中的主体授予访问权限，则不需要额外的基于身份的策略。有关更多信息，请参阅《IAM 用户指南》中的 IAM 中的跨账户资源访问。

基于标签的授权

您可以将标签附加到 Amazon VPC 资源，或者在请求中传递标签。要根据标签控制访问，您需要在策略的 条件元素 中使用条件键来提供标签信息。有关更多信息，请参阅《Amazon EC2 用户指南》中的在创建过程中授予标记资源的权限。

要查看基于身份的策略（用于根据资源上的标签来限制对该资源的访问）的示例，请参阅在特定 VPC 中启动实例。

IAM 角色

IAM 角色是 AWS 账户中具有特定权限的实体。

使用临时凭证

您可以使用临时凭证进行联合身份登录，担任 IAM 角色或担任跨账户角色。您可以通过调用 AWS STS API 操作（如 AssumeRole 或 GetFederationToken）获得临时安全凭证。

Amazon VPC 支持使用临时凭证。

服务相关角色

服务相关角色允许 AWS 服务访问其他服务中的资源以代表您完成操作。服务相关角色显示在 IAM 账户中，并归该服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

中转网关支持服务相关角色。

服务角色

此功能允许服务代表您担任服务角色。此角色允许服务访问其他服务中的资源以代表您完成操作。服务角色显示在 IAM 账户中，并归该账户所有。这意味着，IAM 管理员可以更改该角色的权限。但是，这样做可能会中断服务的功能。

Amazon VPC 支持流日志的服务角色。创建流日志时，您必须选择允许流日志服务访问 CloudWatch Logs 的角色。有关更多信息，请参阅 the section called “用于将流日志发布到 CloudWatch Logs 的 IAM 角色”。

Amazon VPC 策略示例

默认情况下，IAM 角色没有创建或修改 VPC 资源的权限。它们还无法使用 AWS Management Console、AWS CLI 或 AWS API 执行任务。IAM 管理员必须创建 IAM 策略，以便为角色授予权限以对所需的指定资源执行特定的 API 操作。然后，管理员必须将这些策略附加到需要这些权限的 IAM 角色。

要了解如何使用这些示例 JSON 策略文档创建基于 IAM 身份的策略，请参阅《IAM 用户指南》中的创建 IAM 策略。

内容

- 策略最佳实践
- 使用 Amazon VPC 控制台
- 创建带公有子网的 VPC
- 修改和删除 VPC 资源
- 管理安全组
- 管理安全组规则
- 在特定子网中启动实例
- 在特定 VPC 中启动实例
- 屏蔽 VPC 和子网的公共访问权限
- 其他 Amazon VPC 策略示例

策略最佳实践

基于身份的策略确定某个人是否可以创建、访问或删除您账户中的 Amazon VPC 资源。这些操作可能会使 AWS 账户 产生成本。创建或编辑基于身份的策略时，请遵循以下指南和建议：

- AWS 托管式策略及转向最低权限许可入门 – 要开始向用户和工作负载授予权限，请使用 AWS 托管式策略来为许多常见使用场景授予权限。您可以在 AWS 账户 中找到这些策略。我们建议通过定义特定于您的使用场景的 AWS 客户托管式策略来进一步减少权限。有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管式策略或工作职能的 AWS 托管式策略。
- 应用最低权限：在使用 IAM 策略设置权限时，请仅授予执行任务所需的权限。为此，您可以定义在特定条件下可以对特定资源执行的操作，也称为最低权限许可。有关使用 IAM 应用权限的更多信息，请参阅《IAM 用户指南》中的 IAM 中的策略和权限。
- 使用 IAM 策略中的条件进一步限制访问权限：您可以向策略添加条件来限制对操作和资源的访问。例如，您可以编写策略条件来指定必须使用 SSL 发送所有请求。如果通过特定 AWS 服务（例如 AWS CloudFormation）使用服务操作，您还可以使用条件来授予对服务操作的访问权限。有关更多信息，请参阅《IAM 用户指南》中的 IAM JSON 策略元素：条件。
- 使用 IAM Access Analyzer 验证您的 IAM 策略，以确保权限的安全性和功能性 – IAM Access Analyzer 会验证新策略和现有策略，以确保策略符合 IAM 策略语言（JSON）和 IAM 最佳实践。IAM Access Analyzer 提供 100 多项策略检查和可操作的建议，以帮助您制定安全且功能性强的策略。有关更多信息，请参阅《IAM 用户指南》中的使用 IAM Access Analyzer 验证策略。
- 需要多重身份验证（MFA）：如果您所处的场景要求您的 AWS 账户 中有 IAM 用户或根用户，请启用 MFA 来提高安全性。若要在调用 API 操作时需要 MFA，请将 MFA 条件添加到您的策略中。有关更多信息，请参阅《IAM 用户指南》中的使用 MFA 保护 API 访问。

有关 IAM 中的最佳实操的更多信息，请参阅《IAM 用户指南》中的 IAM 中的安全最佳实践。

使用 Amazon VPC 控制台

要访问 Amazon VPC 控制台，您必须具有一组最低的权限。这些权限必须允许您列出和查看有关您 AWS 账户中的 Amazon VPC 资源的详细信息。如果您创建的基于身份的策略比所需的最低权限更严格，则无法为具有该策略的实体（IAM 角色）正常运行控制台。

以下策略授予角色在 VPC 控制台中列出资源的权限，但不允许创建、更新或删除这些资源。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
```

```
                "ec2:DescribeAccountAttributes",
                "ec2:DescribeAddresses",
                "ec2:DescribeAvailabilityZones",
                "ec2:DescribeClassicLinkInstances",
                "ec2:DescribeClientVpnEndpoints",
                "ec2:DescribeCustomerGateways",
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeEgressOnlyInternetGateways",
                "ec2:DescribeFlowLogs",
                "ec2:DescribeInternetGateways",
                "ec2:DescribeManagedPrefixLists",
                "ec2:DescribeMovingAddresses",
                "ec2:DescribeNatGateways",
                "ec2:DescribeNetworkAcls",
                "ec2:DescribeNetworkInterfaceAttribute",
                "ec2:DescribeNetworkInterfacePermissions",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribePrefixLists",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroupReferences",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSecurityGroupRules",
                "ec2:DescribeStaleSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeTags",
                "ec2:DescribeTrafficMirrorFilters",
                "ec2:DescribeTrafficMirrorSessions",
                "ec2:DescribeTrafficMirrorTargets",
                "ec2:DescribeTransitGateways",
                "ec2:DescribeTransitGatewayVpcAttachments",
                "ec2:DescribeTransitGatewayRouteTables",
                "ec2:DescribeVpcAttribute",
                "ec2:DescribeVpcClassicLink",
                "ec2:DescribeVpcClassicLinkDnsSupport",
                "ec2:DescribeVpcEndpoints",
                "ec2:DescribeVpcEndpointConnectionNotifications",
                "ec2:DescribeVpcEndpointConnections",
                "ec2:DescribeVpcEndpointServiceConfigurations",
                "ec2:DescribeVpcEndpointServicePermissions",
                "ec2:DescribeVpcEndpointServices",
                "ec2:DescribeVpcPeeringConnections",
                "ec2:DescribeVpcs",
                "ec2:DescribeVpnConnections",
                "ec2:DescribeVpnGateways",
```

```
                "ec2:GetManagedPrefixListAssociations",
                "ec2:GetManagedPrefixListEntries"
            ],
            "Resource": "*"
        }
    ]
}
```

对于仅调用 AWS CLI 或 AWS API 的角色，您不需要允许最低控制台权限。而应仅允许访问与该角色需要执行的 API 操作相匹配的操作。

创建带公有子网的 VPC

以下示例允许角色创建 VPC、子网、路由表和互联网网关。角色还可以将互联网网关连接到 VPC 并在路由表中创建路由。`ec2:ModifyVpcAttribute` 操作允许角色为 VPC 启用 DNS 主机名，使得在 VPC 中启动的每个实例都会收到一个 DNS 主机名。

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
        "ec2:CreateVpc",
        "ec2:CreateSubnet",
        "ec2:DescribeAvailabilityZones",
        "ec2:CreateRouteTable",
        "ec2:CreateRoute",
        "ec2:CreateInternetGateway",
        "ec2:AttachInternetGateway",
        "ec2:AssociateRouteTable",
        "ec2:ModifyVpcAttribute"
      ],
      "Resource": "*"
    }
   ]
}
```

上述策略还允许角色在 Amazon VPC 控制台中创建 VPC。

修改和删除 VPC 资源

您可能需要控制角色可以修改或删除的 VPC 资源。例如，以下策略允许角色使用和删除具有标签 Purpose=Test 的路由表。该策略还规定角色只能删除具有标签 Purpose=Test 的互联网网关。角色不能使用没有此标签的路由表或互联网网关。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:DeleteInternetGateway",
            "Resource": "arn:aws:ec2:*:*:internet-gateway/*",
            "Condition": {
                "StringEquals": {
                    "ec2:ResourceTag/Purpose": "Test"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DeleteRouteTable",
                "ec2:CreateRoute",
                "ec2:ReplaceRoute",
                "ec2:DeleteRoute"
            ],
            "Resource": "arn:aws:ec2:*:*:route-table/*",
            "Condition": {
                "StringEquals": {
                    "ec2:ResourceTag/Purpose": "Test"
                }
            }
        }
    ]
}
```

管理安全组

以下策略允许角色管理安全组。第一个语句允许角色删除任何带有标签 Stack=test 的安全组，并管理任何带有标签 Stack=test 的安全组的入站和出站规则。第二个语句要求角色使用标签 Stack=Test 来标记其创建的任何安全组。第三个语句允许角色在创建安全组时创建标签。第四个语句允许角色查看任何安全组和安全组规则。第五个语句允许角色在 VPC 中创建安全组。

> **Note**
>
> AWS CloudFormation 服务无法使用此策略来创建带有所需标签的安全组。如果您移除需要标签的 ec2:CreateSecurityGroup 操作的条件，则该策略将生效。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:RevokeSecurityGroupIngress",
                "ec2:AuthorizeSecurityGroupEgress",
                "ec2:AuthorizeSecurityGroupIngress",
                "ec2:UpdateSecurityGroupRuleDescriptionsEgress",
                "ec2:RevokeSecurityGroupEgress",
                "ec2:DeleteSecurityGroup",
                "ec2:ModifySecurityGroupRules",
                "ec2:UpdateSecurityGroupRuleDescriptionsIngress"
            ],
            "Resource": "arn:aws:ec2:*:*:security-group/*",
            "Condition": {
                "StringEquals": {
                    "ec2:ResourceTag/Stack": "test"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateSecurityGroup",
            "Resource": "arn:aws:ec2:*:*:security-group/*",
```

```
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/Stack": "test"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": "Stack"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "arn:aws:ec2:*:*:security-group/*",
            "Condition": {
                "StringEquals": {
                    "ec2:CreateAction": "CreateSecurityGroup"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeSecurityGroupRules",
                "ec2:DescribeVpcs",
                "ec2:DescribeSecurityGroups"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateSecurityGroup",
            "Resource": "arn:aws:ec2:*:*:vpc/*"
        }
    ]
}
```

要允许角色更改与实例关联的安全组，请将 ec2:ModifyInstanceAttribute 操作添加到策略中。

要允许角色能够更改网络接口的安全组，请将 ec2:ModifyNetworkInterfaceAttribute 操作添加到策略中。

管理安全组规则

以下策略为角色授予相应的权限，以便查看所有安全组和安全组规则、添加和移除特定 VPC 的安全组的入站和出站规则，以及为指定 VPC 修改规则描述。第一个语句使用 `ec2:Vpc` 条件键来将权限范围扩展到特定 VPC。

第二个语句授予角色描述所有安全组、安全组规则和标签的权限。这使角色能够查看安全组规则以便对其进行修改。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AuthorizeSecurityGroupIngress",
                "ec2:RevokeSecurityGroupIngress",
                "ec2:UpdateSecurityGroupRuleDescriptionsIngress",
                "ec2:AuthorizeSecurityGroupEgress",
                "ec2:RevokeSecurityGroupEgress",
                "ec2:UpdateSecurityGroupRuleDescriptionsEgress",
                "ec2:ModifySecurityGroupRules"
            ],
            "Resource": "arn:aws:ec2:us-east-1:123456789012:security-group/*",
            "Condition": {
                "ArnEquals": {
                    "ec2:Vpc": "arn:aws:ec2:us-
east-1:123456789012:vpc/vpc-1234567890abcdef0"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSecurityGroupRules",
                "ec2:DescribeTags"
            ],
            "Resource": "*"
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "ec2:ModifySecurityGroupRules"
            ],
            "Resource": "arn:aws:ec2:us-east-1:123456789012:security-group-rule/
*"
        }
    ]
}
```

在特定子网中启动实例

以下策略允许角色在特定子网中启动实例，以及在请求中使用特定安全组。该策略通过指定子网 ARN 和安全组 ARN 来实现上述目的。如果角色尝试在其他子网中启动实例或使用其他的安全组，请求将失败（除非其他策略或语句授予角色相应的权限）。

该策略还授予使用网络接口资源的权限。在子网中启动时，RunInstances 请求会默认创建一个主网络接口，因此角色在启动实例时需要有创建此资源的权限。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:us-east-1::image/ami-*",
                "arn:aws:ec2:us-east-1:123456789012:instance/*",
                "arn:aws:ec2:us-
east-1:123456789012:subnet/subnet-1234567890abcdef0",
                "arn:aws:ec2:us-east-1:123456789012:network-interface/*",
                "arn:aws:ec2:us-east-1:123456789012:volume/*",
                "arn:aws:ec2:us-east-1:123456789012:key-pair/*",
                "arn:aws:ec2:us-east-1:123456789012:security-
group/sg-0abcdef1234567890"
            ]
        }
    ]
```

```
}
```

在特定 VPC 中启动实例

以下策略允许角色在特定 VPC 中的任意子网中启动实例。此策略通过将条件密钥 (ec2:Vpc) 应用于子网资源来实现上述目的。

该策略还授予角色仅使用具有标签“department=dev”的 AMI 启动实例的权限。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:us-east-1:123456789012:subnet/*",
            "Condition": {
                "ArnEquals": {
                    "ec2:Vpc": "arn:aws:ec2:us-
east-1:123456789012:vpc/vpc-1234567890abcdef0"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
            "Condition": {
                "StringEquals": {
                    "ec2:ResourceTag/department": "dev"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:us-east-1:123456789012:instance/*",
                "arn:aws:ec2:us-east-1:123456789012:volume/*",
                "arn:aws:ec2:us-east-1:123456789012:network-interface/*",
```

```
                "arn:aws:ec2:us-east-1:123456789012:key-pair/*",
                "arn:aws:ec2:us-east-1:123456789012:security-group/*"
            ]
        }
    ]
}
```

屏蔽 VPC 和子网的公共访问权限

以下策略示例授予角色使用 VPC 屏蔽公共访问权限（BPA）功能的权限，以屏蔽 VPC 和子网中资源的公共访问权限。

示例 1：允许对 VPC BPA 账户范围的设置和 VPC BPA 排除项的只读访问。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VPCBPAReadOnlyAccess",
      "Action": [
        "ec2:DescribeVpcBlockPublicAccessOptions",
        "ec2:DescribeVpcBlockPublicAccessExclusions"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

示例 2：允许对 VPC BPA 账户范围的设置和 VPC BPA 排除项的完全读取和写入访问。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```
      "Sid": "VPCBPAFullAccess",
      "Action": [
        "ec2:DescribeVpcBlockPublicAccessOptions",
        "ec2:DescribeVpcBlockPublicAccessExclusions",
        "ec2:ModifyVpcBlockPublicAccessOptions",
        "ec2:CreateVpcBlockPublicAccessExclusion",
        "ec2:ModifyVpcBlockPublicAccessExclusion",
        "ec2:DeleteVpcBlockPublicAccessExclusion"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

示例 3：允许访问所有 EC2 API，但修改 VPC BPA 设置和创建排除项除外。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EC2FullAccess",
            "Action": [
                "ec2:*"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Sid": "VPCBPAPartialAccess",
            "Action": [
                "ec2:ModifyVpcBlockPublicAccessOptions",
                "ec2:CreateVpcBlockPublicAccessExclusion"
            ],
            "Effect": "Deny",
            "Resource": "*"
        }
    ]
}
```

其他 Amazon VPC 策略示例

您可以在以下文档中找到与 Amazon VPC 相关的其他示例 IAM 策略：

- 托管前缀列表
- 流量镜像
- 中转网关
- VPC 端点和 VPC 端点服务 (AWS PrivateLink)
- VPC 对等连接

Amazon VPC 身份和访问疑难解答

可以使用以下信息，以帮助您诊断和修复在使用 Amazon VPC 和 IAM 时可能遇到的常见问题。

问题

- 我无权在 Amazon VPC 中执行操作
- 我无权执行 iam:PassRole
- 我希望允许我的 AWS 账户以外的人访问我的 Amazon VPC 资源

我无权在 Amazon VPC 中执行操作

如果 AWS Management Console 告诉您，您无权执行某个操作，则必须联系您的管理员寻求帮助。管理员是向您提供登录凭证的人。

当 `mateojackson` IAM 用户尝试使用控制台查看有关某个子网的详细信息，但该子网属于不具有 `ec2:DescribeSubnets` 权限的 IAM 角色时，则会发生以下示例错误。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 ec2:DescribeSubnets on resource: subnet-id
```

在这种情况下，Mateo 需要请求管理员更新策略，以允许他访问该子网。

我无权执行 iam:PassRole

如果您收到一个错误，表明您无权执行 `iam:PassRole` 操作，则必须更新策略以允许您将角色传递给 Amazon VPC。

有些 AWS 服务 允许将现有角色传递到该服务，而不是创建新服务角色或服务相关角色。为此，您必须具有将角色传递到服务的权限。

当名为 marymajor 的 IAM 用户尝试使用控制台在 Amazon VPC 中执行操作时，会发生以下示例错误。但是，服务必须具有服务角色所授予的权限才可执行此操作。Mary 不具有将角色传递到服务的权限。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

在这种情况下，必须更新 Mary 的策略以允许她执行 iam:PassRole 操作。

如果您需要帮助，请联系 AWS 管理员。您的管理员是提供登录凭证的人。

我希望允许我的 AWS 账户以外的人访问我的 Amazon VPC 资源

您可以创建一个角色，以便其他账户中的用户或您组织外的人员可以使用该角色来访问您的资源。您可以指定谁值得信赖，可以代入角色。对于支持基于资源的策略或访问控制列表（ACL）的服务，您可以使用这些策略向人员授予对您的资源的访问权。

要了解更多信息，请参阅以下内容：

- 要了解 Amazon VPC 是否支持这些功能，请参阅 Amazon VPC 如何与 IAM 配合使用。
- 要了解如何为您拥有的 AWS 账户中的资源提供访问权限，请参阅《IAM 用户指南》中的为您拥有的另一个 AWS 账户中的 IAM 用户提供访问权限。
- 要了解如何为第三方 AWS 账户 提供您的资源的访问权限，请参阅《IAM 用户指南》中的为第三方拥有的 AWS 账户 提供访问权限。
- 要了解如何通过身份联合验证提供访问权限，请参阅《IAM 用户指南》中的为经过外部身份验证的用户（身份联合验证）提供访问权限。
- 要了解使用角色和基于资源的策略进行跨账户访问之间的差别，请参阅《IAM 用户指南》中的 IAM 中的跨账户资源访问。

适用于 Amazon Virtual Private Cloud 的 AWS 托管策略

AWS 托管式策略是由 AWS 创建和管理的独立策略。AWS 托管式策略旨在为许多常见使用场景提供权限，以便您可以开始为用户、组和角色分配权限。

请记住，AWS 托管策略可能不会为您的特定使用场景授予最低权限，因为它们可供所有 AWS 客户使用。我们建议通过定义特定于您的使用场景的客户管理型策略来进一步减少权限。

您无法更改 AWS 托管策略中定义的权限。如果 AWS 更新在 AWS 托管策略中定义的权限，则更新会影响该策略所附加到的所有主体身份（用户、组和角色）。当新的 AWS 服务 启动或新的 API 操作可用于现有服务时，AWS 最有可能更新 AWS 托管策略。

有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管策略。

AWS 托管策略：AmazonVPCFullAccess

您可以将 `AmazonVPCFullAccess` 策略附加得到 IAM 身份。此策略授予允许完全访问 Amazon VPC 的权限。

要查看此策略的权限，请参阅《AWS 托管式策略参考》中的 AmazonVPCFullAccess。

AWS 托管策略：AmazonVPCReadOnlyAccess

您可以将 `AmazonVPCReadOnlyAccess` 策略附加得到 IAM 身份。此策略授予允许对 Amazon VPC 进行只读访问的权限。

要查看此策略的权限，请参阅《AWS 托管式策略参考》中的 AmazonVPCReadOnlyAccess。

AWS 托管式策略: AmazonVPCCrossAccountNetworkInterfaceOperations

您可以将 `AmazonVPCCrossAccountNetworkInterfaceOperations` 策略附加到 IAM 身份。此策略授予可权限，允许身份创建网络接口并将其附加到跨账户资源。

要查看此策略的权限，请参阅《AWS 托管式策略参考》中的 AmazonVPCCrossAccountNetworkInterfaceOperations。

AWS 托管策略的 Amazon VPC 更新

查看有关 Amazon VPC 的 AWS 托管策略的更新的详细信息（此服务于 2021 年 3 月开始跟踪这些更改）。

更改	描述	日期
the section called “AmazonVPCFullAccess” – 对现有策略的更新	添加了 AssociateSecurityGroupVpc、DescribeSecurityGroupVpcAssociations 和 DisassociateSecurityGroupVpc 操作，允许您关	2024 年 12 月 9 日

更改	描述	日期
	联、取消关联和查看安全组与 VPC 的关联。	
the section called “AmazonVPCReadOnlyAccess” – 对现有策略的更新	添加了 DescribeSecurityGroupVpcAssociations 操作，允许您查看安全组与 VPC 的关联。	2024 年 12 月 9 日
the section called “AmazonVPCFullAccess” – 对现有策略的更新	添加了 GetSecurityGroupsForVpc 操作，允许您获取可在 VPC 中使用的安全组。	2024 年 2 月 8 日
the section called “AmazonVPCReadOnlyAccess” – 对现有策略的更新	添加了 GetSecurityGroupsForVpc 操作，允许您获取可在 VPC 中使用的安全组。	2024 年 2 月 8 日
the section called “AmazonVPCCrossAccountNetworkInterfaceOperations” – 对现有策略的更新	添加 AssignIpv6Addresses 和 UnassignIpv6Addresses 操作，允许您管理与网络接口关联的 IPv6 地址。	2023 年 9 月 25 日
the section called “AmazonVPCReadOnlyAccess” – 对现有策略的更新	添加了 DescribeSecurityGroupRules 操作，可查看安全组规则。	2021 年 8 月 2 日
the section called “AmazonVPCFullAccess” – 对现有策略的更新	添加了 DescribeSecurityGroupRules 和 ModifySecurityGroupRules 操作，可查看和修改安全组规则。	2021 年 8 月 2 日
the section called “AmazonVPCFullAccess” – 对现有策略的更新	添加了针对运营商网关、IPv6 池、本地网关和本地网关路由表的操作。	2021 年 6 月 23 日
the section called “AmazonVPCReadOnlyAccess” – 对现有策略的更新	添加了针对运营商网关、IPv6 池、本地网关和本地网关路由表的操作。	2021 年 6 月 23 日

Amazon VPC 中的基础设施安全性

作为一项托管服务，Amazon Virtual Private Cloud 享受 AWS 全球网络安全的保护。有关 AWS 安全服务以及 AWS 如何保护基础设施的信息，请参阅 AWS 云安全性。要按照基础设施安全最佳实践设计您的 AWS 环境，请参阅《安全性支柱 AWS Well-Architected Framework》中的 基础设施保护。

您可以使用 AWS 发布的 API 调用通过网络访问 Amazon VPC。客户端必须支持以下内容：

- 传输层安全性协议（TLS）。我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 具有完全向前保密（PFS）的密码套件，例如 DHE（临时 Diffie-Hellman）或 ECDHE（临时椭圆曲线 Diffie-Hellman）。大多数现代系统（如 Java 7 及更高版本）都支持这些模式。

此外，必须使用访问密钥 ID 和与 IAM 主体关联的秘密访问密钥来对请求进行签名。或者，您可以使用 AWS Security Token Service（AWS STS）生成临时安全凭证来对请求进行签名。

网络隔离

虚拟私有云（VPC）是 AWS 云内您自己的逻辑隔离区域中的虚拟网络。可以使用单独的 VPC 按工作负载或组织实体隔离基础设施。

子网是 VPC 中的 IP 地址范围。在启动实例时，您可以在 VPC 上的子网中启动该实例。可以使用子网隔离单个 VPC 中的应用程序层（例如，Web、应用程序和数据库）。如果不应直接从 Internet 访问实例，请使用私有子网访问。

您可以使用 AWS PrivateLink，来让 VPC 中的资源可以使用私有 IP 地址连接到 AWS 服务，如同这些服务直接在您的 VPC 中托管。因此，您无需使用互联网网关或 NAT 设备即可访问 AWS 服务。

控制网络流量

请考虑使用以下方法来控制到 VPC 中资源（例如 EC2 实例）的网络流量：

- 将 安全组 作为主要机制来控制对 VPC 的网络访问。必要时可使用 网络 ACL 来实现无状态的粗略网络控制。安全组能够执行有状态数据包筛选，以及创建引用其他安全组的规则，因此其功能比网络 ACL 更丰富。将网络 ACL 作为辅助控制机制（例如拒绝特定的流量子集），或者作为高级别的子网防护机制时，也可能非常有效。此外，由于网络 ACL 将应用于整个子网，因此可以用作深度防御机制，以防实例在没有正确配置安全组的情况下启动。
- 如果不应直接从 Internet 访问实例，请使用私有子网访问。使用堡垒主机或 NAT 网关从私有子网中的实例访问互联网。

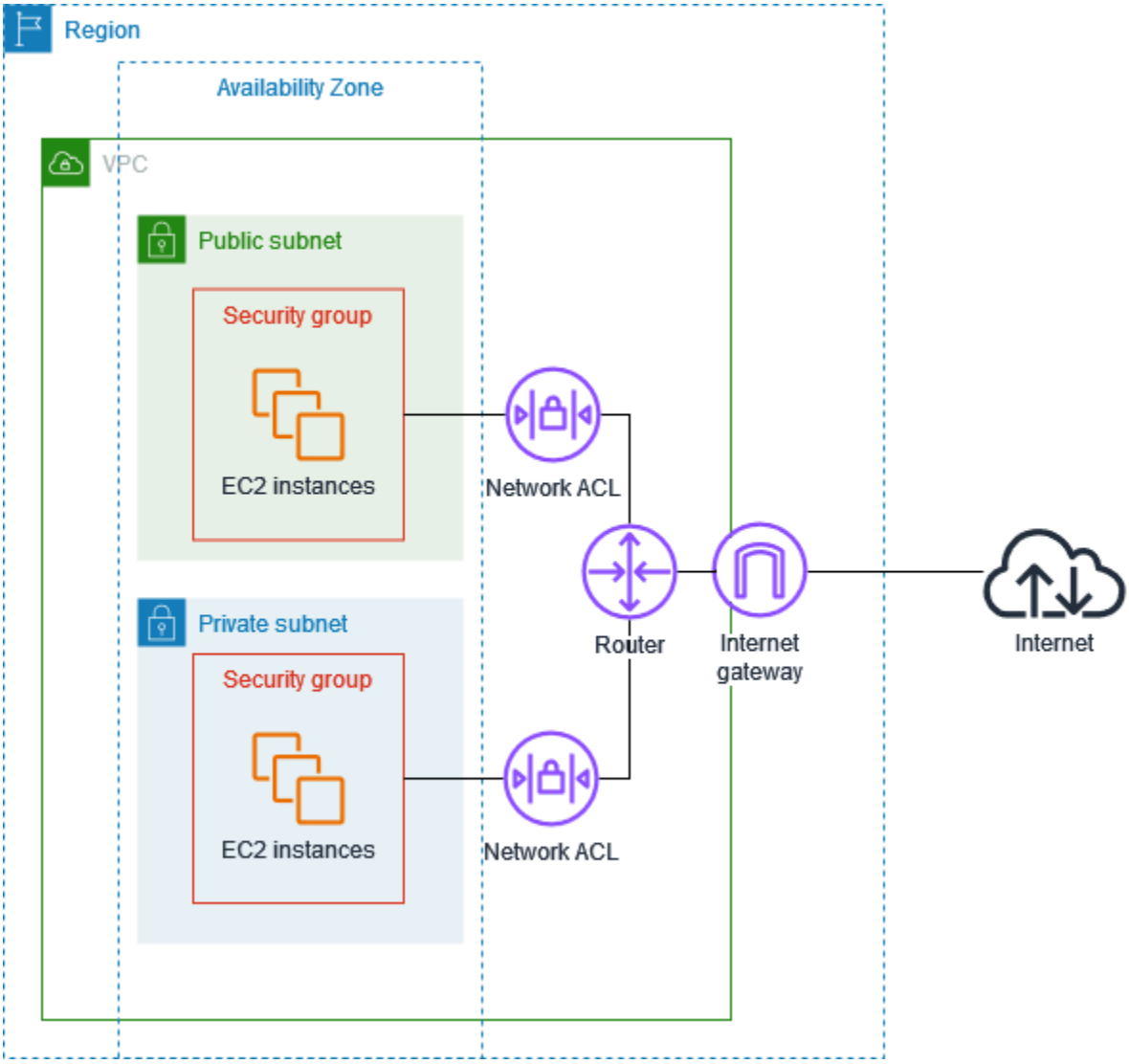
- 配置子网路由表，其中应包含能够支持连接需求的最低网络路由。
- 考虑使用其他安全组或网络接口，独立于常规应用程序流量来控制和审计 Amazon EC2 实例管理流量。这样，您可以实施用于更改控制的特殊 IAM 策略，从而更轻松地审计对安全组规则或自动化规则验证脚本进行的更改。此外，使用多个网络接口还提供了额外的网络流量控制选择，包括能够创建基于主机的路由策略或利用基于分配到子网的网络接口的其他 VPC 子网路由规则。
- 使用 AWS Virtual Private Network 或 AWS Direct Connect 建立从远程网络到 VPC 的私有连接。有关更多信息，请参阅网络到 Amazon VPC 的连接选项。
- 使用 VPC Flow Logs 监控到达实例的流量。
- 使用 AWS Security Hub 检查来自实例的意外网络访问。
- 使用 AWS Network Firewall 来保护 VPC 中的子网，防止常见的网络威胁。

比较安全组和网络 ACL

下表概述了安全组和网络 ACL 之间的基本差异。

特征	安全组	网络 ACL
操作级别	实例级别	子网级别
范围	适用于与安全组关联的所有实例	适用于关联子网中的所有实例
Rule type	仅允许规则	允许和拒绝规则
规则评估	在决定是否允许流量前评估所有规则	按升序评估规则，直到找到与流量匹配的规则
返回流量	自动允许（有状态）	必须明确允许（无状态）

下图展示了由安全组和网络 ACL 提供的安全层。例如，来自互联网网关的数据流会使用路由表中的路由，路由到合适的子网。与子网关联的网络 ACL 规则控制允许进入子网的数据流。与实例关联的安全组规则控制允许进入实例的数据流。



您只能使用安全组保护您的实例。但是，您可以添加网络 ACL 作为额外的防御层。有关更多信息，请参阅 示例：控制对子网中的实例的访问。

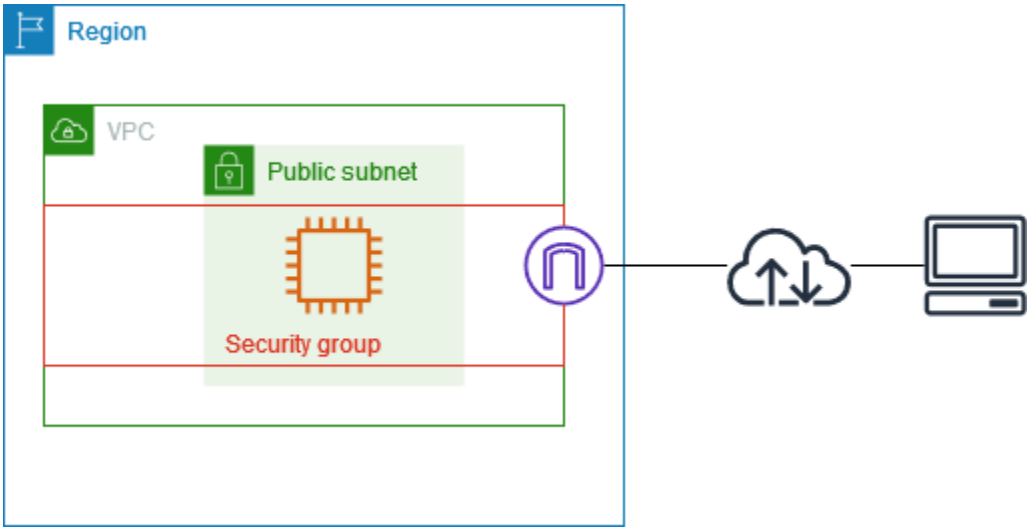
使用安全组控制指向 AWS 资源的流量

安全组控制允许到达和离开与其关联资源的流量。例如，在将安全组与 EC2 实例关联后，它将控制该实例的入站和出站流量。

创建 VPC 时，它带有一个默认安全组。您可以为 VPC 创建额外的安全组，每个安全组都具有自己的入站和出站规则。您可以针对每条入站规则指定来源、端口范围和协议。您可以针对每条出站规则指定目的地、端口范围和协议。

下图显示了一个具有一个子网、一个互联网网关和一个安全组的 VPC。子网包含了一个 EC2 实例。将安全组分配给实例。安全组起到虚拟防火墙的作用。到达实例的唯一流量是得到安全组规则允许的流

量。例如，如果安全组包含一条允许从您的网络到实例的 ICMP 流量的规则，则您可以从您的计算机对该实例进行 ping 操作。如果安全组未包含允许 SSH 流量的规则，则您无法使用 SSH 连接到您的实例。



内容

- 安全组基本信息
- 安全组示例
- 安全组规则
- 您的 VPC 的默认安全组
- 为 VPC 创建安全组
- 配置安全组规则
- 删除安全组
- 将安全组与多个 VPC 关联
- 与 AWS Organizations 共享安全组

定价

使用安全组不收取任何额外费用。

安全组基本信息

- 如果使用安全组 VPC 关联功能将安全组关联到同一区域中的其他 VPC，则可以将安全组分配给与安全组在同一 VPC 中创建的资源或其他 VPC 中的资源。您还可以为单个资源分配多个安全组。
- 创建安全组时，您必须为其提供名称和描述。以下规则适用：

  - 安全组名称在 VPC 中必须是唯一的。
  - 安全组名称不区分大小写。
  - 名称和描述的长度最多为 255 个字符。
  - 名称和描述只能使用以下字符：a-z、A-Z、0-9、空格和 ._-:/()#,@[]+=&;{}!$*。
  - 如果名称后面带有空格，我们在保存名称时会删除这些空格。例如，如果您输入“Test Security Group”作为名称，我们会将其存储为“Test Security Group”。
  - 安全组名称不能以 sg- 开头。
- 安全组是有状态的。例如，如果您从实例发送请求，则无论入站安全组规则如何，都允许该请求的响应流量到达该实例。如果是为响应已允许的入站流量，则该响应可以离开实例，此时可忽略出站规则。
- 安全组不会筛选发往和来自以下位置的流量：
  - Amazon 域名服务 (DNS)
  - Amazon 动态主机配置协议 (DHCP)
  - Amazon EC2 实例元数据
  - Amazon ECS 任务元数据端点
  - Windows 实例的许可证激活
  - Amazon Time Sync Service
  - 默认 VPC 路由器使用的预留 IP 地址
- 系统对您为每个 VPC 创建的安全组数、向每个安全组添加的规则数以及与网络接口关联的安全组数设有配额。有关更多信息，请参阅 Amazon VPC 配额。

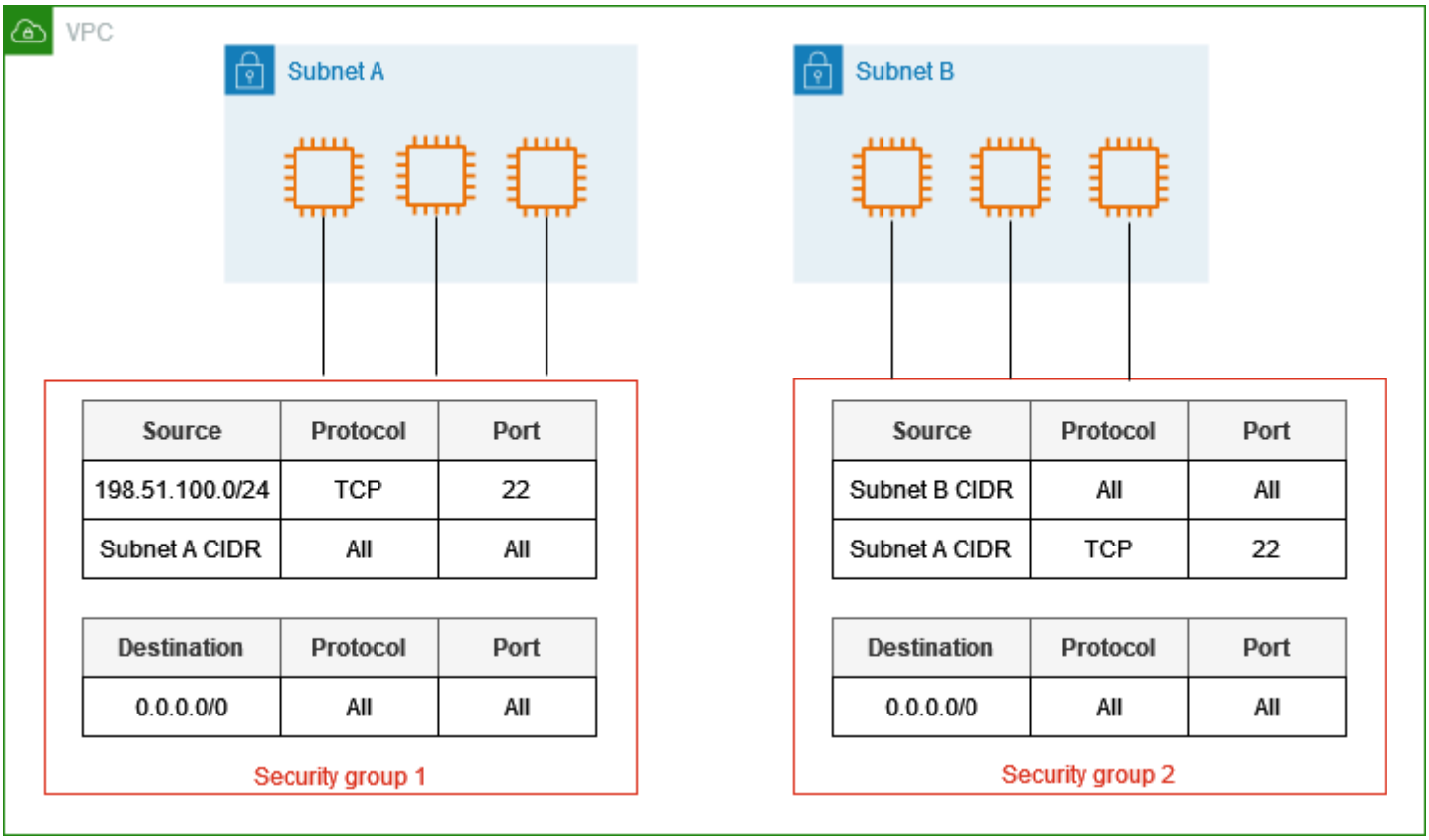
最佳实践

- 仅授权特定 IAM 主体创建和修改安全组。
- 创建所需的最小数量的安全组，以降低出错风险。使用每个安全组，对具有相似功能和安全要求的资源的访问权限进行管理。
- 在为端口 22（SSH）或 3389（RDP）添加入站规则以便访问 EC2 实例时，仅授权特定 IP 地址范围。如果您指定 0.0.0.0/0（IPv4）和 ::/（IPv6），则任何人都可以使用指定协议从任何 IP 地址访问您的实例。
- 请勿设置较大端口范围。确保通过每个端口的访问仅限于需要访问的源或目的地。
- 您可以考虑建立网络 ACL，使其规则与您的安全组规则相似，以便为 VPC 添加额外安全层。有关安全组和网络 ACL 之间的差别的更多信息，请参见 比较安全组和网络 ACL。

安全组示例

以下示意图显示了具有两个安全组和两个子网的 VPC。子网 A 中的实例具有相同的连接要求，因此这些实例与安全组 1 相关联。子网 B 中的实例具有相同的连接要求，因此这些实例与安全组 2 相关联。安全组规则允许流量通过，如下所示：

- 安全组 1 中的第一条入站规则允许从指定地址范围（例如，您自己网络中的范围）到子网 A 中实例的 SSH 流量。
- 安全组 1 中的第二条入站规则允许子网 A 中的实例使用任何协议和端口进行相互通信。
- 安全组 2 中的第一条入站规则允许子网 B 中的实例使用任何协议和端口进行相互通信。
- 安全组 2 中的第二条入站规则允许子网 A 中的实例使用 SSH 与子网 B 中的实例进行通信。
- 两个安全组都使用默认出站规则，以允许所有流量。



安全组规则

安全组的规则控制允许达到与该安全组相关联资源的入站流量。这些规则还控制允许离开实例的出站流量。

您可以添加或删除安全组规则（又被称为授权或撤销入站或出站访问）。适用于入站数据流（进入）或出站数据流（离开）的规则。您可以授予对特定源或目标的访问权限。

内容

- 安全组规则基本信息
- 安全组规则的组成部分
- 引用安全组
- 安全组大小
- 过时的安全组规则

安全组规则基本信息

以下是您的安全组规则的特征：

- 您可以指定允许规则，但不可指定拒绝规则。
- 当您首次创建安全组时，它没有入站规则。因此，在将入站规则添加到安全组之前，不允许入站流量。
- 首次创建安全组时，它具有允许来自资源的所有出站流量的出站规则。您可以删除该规则并添加只允许特定出站流量的出站规则。如果您的安全组没有出站规则，则不允许出站流量。
- 当您将多个安全组与一个资源相关联时，来自每个安全组的规则将聚合形成一组规则，用于确定是否允许访问。
- 添加、更新或删除规则时，您的更改会自动应用于与安全组关联的所有资源。有关说明，请参阅配置安全组规则。
- 某些规则变更产生的影响可能会取决于跟踪流量的方式。有关更多信息，请参阅《Amazon EC2 用户指南》中的连接跟踪。
- 当您创建安全组规则时，AWS 会将唯一 ID 分配给规则。当您使用 API 或 CLI 修改或删除某规则时，您可以使用该规则的 ID。

限制

安全组无法阻止发送至 Route 53 Resolver 或来自其的 DNS 请求，Route 53 Resolver 有时称为“VPC +2 IP 地址”（请参阅《Amazon Route 53 开发人员指南》中的 Amazon Route 53 Resolver）或 AmazonProvidedDNS。要通过 Route 53 Resolver 筛选 DNS 请求，请使用 Route 53 Resolver DNS Firewall。

安全组规则的组成部分

以下是安全组入站和出站规则的组件：

- 协议：允许的协议。最常见的协议为 6 (TCP)、17 (UDP) 和 1 (ICMP)。
- 端口范围：对于 TCP、UDP 或自定义协议，允许的端口范围。您可以指定单个端口号（例如 22）或端口号范围（例如`7000-8000`）。
- ICMP 类型和代码：对于 ICMP，ICMP 类型和代码。例如，对于 ICMP 回应请求使用类型 8，对 ICMPv6 回显请求使用键入 128。有关更多信息，请参阅《Amazon EC2 用户指南》中的用于 ping/ICMP 的规则。
- 源或目标：允许的流量的源（入站规则）或目标（出站规则）。指定下列项之一：
  - 一个 IPv4 地址。您必须使用 `/32` 前缀长度。例如 `203.0.113.1/32`。
  - 一个 IPv6 地址。您必须使用 `/128` 前缀长度。例如 `2001:db8:1234:1a00::123/128`。
  - 采用 CIDR 块表示法的 IPv4 地址范围。例如 `203.0.113.0/24`。
  - 采用 CIDR 块表示法的 IPv6 地址范围。例如 `2001:db8:1234:1a00::/64`。
  - 前缀列表的 ID。例如 `pl-1234abc1234abc123`。有关更多信息，请参阅 托管前缀列表。
  - 安全组的 ID。例如 `sg-1234567890abcdef0`。有关更多信息，请参阅 the section called “引用安全组”。
- （可选）描述：您可以添加规则的说明；这可帮助您在以后识别它。描述的长度最多为 255 个字符。允许的字符包括 a-z、A-Z、0-9、空格和 ._-:/()#,@[]+=;{}!$*。

有关示例，请参阅《Amazon EC2 用户指南》中针对不同用例的安全组规则。

引用安全组

当您指定一个安全组作为规则的源或目标时，该规则会影响与安全组关联的所有实例。实例可以使用其私有 IP 地址，通过指定的协议和端口沿指定方向进行通信。

例如，下面的内容表示安全组的入站规则，该入站规则引用了安全组 sg-0abcdef1234567890。此规则允许来自与 sg-0abcdef1234567890 关联的实例的入站 SSH 流量。

来源	协议	端口范围
sg-0abcdef1234567890	TCP	22

在安全组规则中引用安全组时，请注意以下几点：

- 如果满足以下任一条件，则可以在其他安全组的入站规则中引用该安全组：
  - 与同一 VPC 关联的安全组。
  - 与安全组关联的 VPC 之间存在对等连接。
  - 与安全组关联的 VPC 之间存在中转网关。
- 如果满足以下任一条件，则可以在出站规则中引用安全组：
  - 与同一 VPC 关联的安全组。
  - 与安全组关联的 VPC 之间存在对等连接。
- 不得向引用安全组的安全组添加引用安全组中的任何规则。
- 对于入站规则，与安全组关联的 EC2 实例可以接收来自与引用安全组关联的 EC2 实例的私有 IP 地址的入站流量。
- 对于出站规则，与安全组关联的 EC2 实例可以向与引用安全组关联的 EC2 实例的私有 IP 地址发送出站流量。

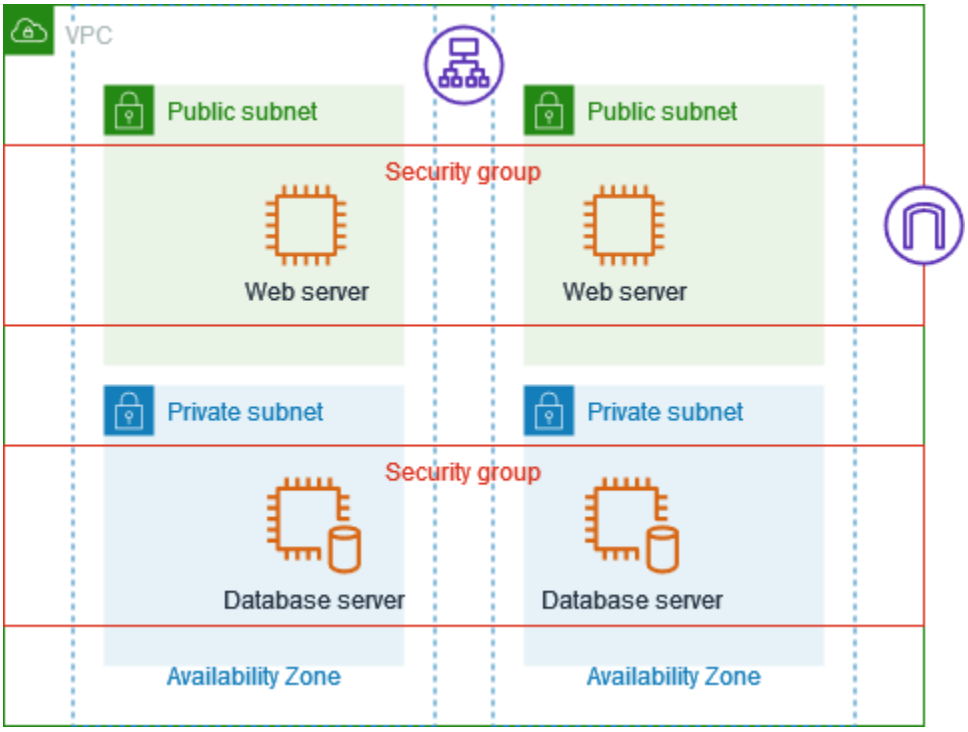
限制

如果您将路由配置为通过中间设备在不同子网中的两个实例之间转发流量，则必须确保这两个实例的安全组允许流量在实例之间流动。每个实例的安全组必须引用另一个实例的私有 IP 地址或包含另一个实例的子网的 CIDR 范围作为源。如果您引用另一个实例的安全组作为源，则安全组不允许流量在实例之间流动。

示例

下图显示了一个在两个可用区内分布子网，此外还拥有一个互联网网关和一个应用程序负载均衡器的 VPC。每个可用区都有一个用于 Web 服务器的公有子网和一个用于数据库服务器的私有子网。负载均衡器、Web 服务器和数据库服务器有单独的安全组。创建以下安全组规则以允许流量。

- 向负载均衡器安全组添加规则，以允许来自互联网的 HTTP 和 HTTPS 流量。来源是 0.0.0.0/0。
- 向 Web 服务器的安全组添加规则，以仅允许来自负载均衡器的 HTTP 和 HTTPS 流量。来源是负载均衡器的安全组。
- 向数据库服务器的安全组添加规则，以允许来自 Web 服务器的数据库请求。来源是 Web 服务器的安全组。



安全组大小

源或目标的类型决定了将每条规则计入每个安全组可以拥有的最大规则数量的方式。

- 引用 CIDR 块的规则计为一条规则。
- 无论引用的安全组大小如何，引用其他安全组的规则均计为一条规则。
- 引用客户托管式前缀列表的规则计为前缀列表的最大大小。例如，如果前缀列表的最大大小为 20，则引用此前缀列表的规则计为 20 条规则。
- 引用 AWS 托管式前缀列表的规则计为前缀列表的权重。例如，如果前缀列表的权重为 10，则引用此前缀列表的规则计为 10 条规则。有关更多信息，请参阅 the section called “可用的 AWS 托管前缀列表”。

过时的安全组规则

如果您的 VPC 具有与其他 VPC 的 VPC 对等连接，或者如果它使用其他账户共享的 VPC，则您的 VPC 中的安全组规则可引用该对等 VPC 或共享 VPC 中的安全组。这样，与所引用安全组关联的资源以及与进行引用的安全组关联的资源可以相互通信。有关更多信息，请参阅《Amazon VPC 对等指南》中的更新您的安全组以引用对等安全组。

如果安全组规则引用了对等 VPC 或共享 VPC 中的安全组，而共享 VPC 中的安全组或 VPC 对等连接已被删除，则该安全组规则将会标记为过时。与任何其他的安全组规则一样，您可以删除过时的安全组规则。

您的 VPC 的默认安全组

您的默认 VPC 和您创建的任何 VPC 都带有默认安全组。默认安全组的名称为“default”。

建议您为特定资源或资源组创建安全组，而不要使用默认安全组。然而如果您在创建时未将某些资源关联到安全组，我们会将其关联到默认安全组。例如，如果您在启动 EC2 实例时未指定安全组，则我们会将该实例关联到 VPC 的默认安全组。

默认安全组基本信息

- 您可以更改默认安全组的规则。
- 您无法删除默认安全组。如果您尝试删除默认安全组，会显示以下错误代码：`Client.CannotDelete`。

默认规则

下表介绍默认安全组的默认入站规则。

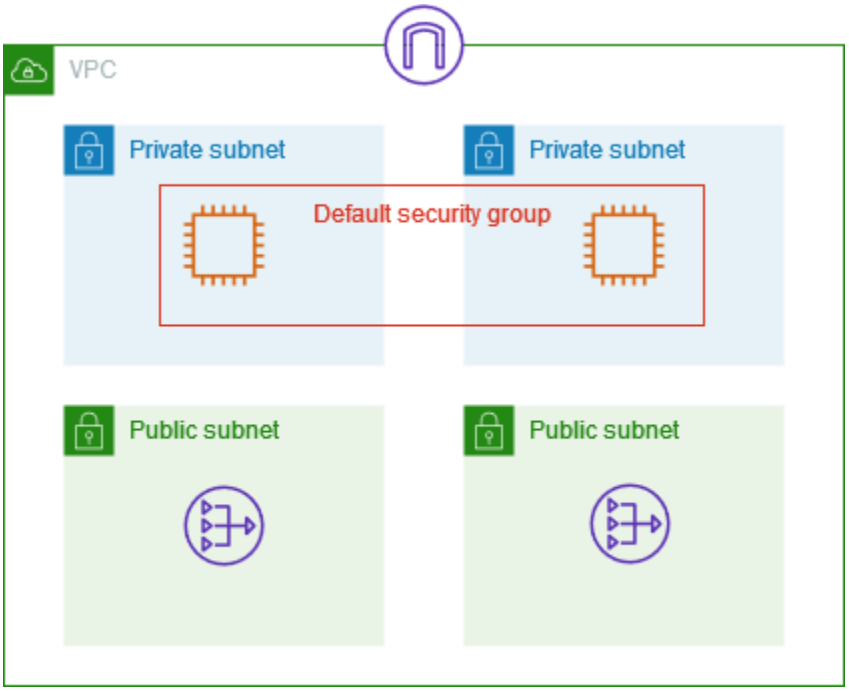
来源	协议	端口范围	描述
sg-1234567890abcdef0	全部	全部	允许来自分配给此安全组的所有资源的入站流量。源为此安全组的 ID。

下表介绍默认安全组的默认出站规则。

目标	协议	端口范围	描述
0.0.0.0/0	All	All	允许所有的出站 IPv4 流量。
::/0	All	All	允许所有的出站 IPv6 流量。仅当 VPC 具有关联的 IPv6 CIDR 块时才添加此规则。

示例

下图显示了一个具有一个默认安全组、一个互联网网关和一个 NAT 网关的 VPC。默认安全组仅包含其默认规则，并且与在 VPC 中运行的两个 EC2 实例相关联。在此场景中，每个实例都可以在所有端口和协议上接收来自另一个实例的入站流量。默认规则不允许这些实例接收来自互联网网关或 NAT 网关的流量。如果您的实例必须接收更多流量，则建议您使用所需规则创建安全组，并将新的安全组与实例关联，而不是与默认安全组。



为 VPC 创建安全组

您的虚拟私有云（VPC）带有默认的安全组。您可以创建额外的安全组。安全组只能与创建它的 VPC 中的资源结合使用。

在默认情况下，新安全组起初只有一条出站规则，即允许所有通信离开资源。您必须添加规则，以便允许任何入站数据流或限制出站数据流。您可以在创建安全组时添加安全组规则，也可以稍后再添加。有关更多信息，请参阅 安全组规则。

所需的权限

在开始之前，请确保您拥有所需权限。有关更多信息，请参阅下列内容：

- 管理安全组
- 管理安全组规则

使用控制台创建安全组

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择安全组。
3. 选择Create security group（创建安全组）。
4. 输入安全组的名称和描述。在创建安全组后，您无法更改其名称和描述。
5. 对于 VPC，请选择要在其中创建资源并将其与安全组关联的 VPC。
6. （可选）要添加入站规则，请选择入站规则。对于每条规则，请选择添加规则并指定协议、端口和来源。有关更多信息，请参阅 配置安全组规则。
7. （可选）要添加出站规则，请选择出站规则。对于每条规则，请选择添加规则并指定协议、端口和目标。
8. （可选）若要添加标签，请选择 Add new tag（添加新标签），然后输入该标签的键和值。
9. 选择 Create security group（创建安全组）。

使用 AWS CLI 创建安全组

使用 create-security-group 命令。

此外，您可以通过复制现有安全组来创建新安全组。复制安全组时，我们会自动添加与原始安全组相同的入站和出站规则，也会使用与原始安全组相同的 VPC。您可以为新安全组输入名称和描述。您可以选择另一个 VPC，也可以根据需要修改入站和出站规则。不过，您无法将安全组从一个区域复制到另一区域。

根据现有安全组创建安全组

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择安全组。
3. 选择一个安全组。
4. 依次选择操作、复制到新安全组。
5. 输入安全组的名称和描述。
6. （可选）根据需要选择另一个 VPC。
7. （可选）根据需要添加、删除或编辑安全组规则。
8. 选择 Create security group（创建安全组）。

配置安全组规则

创建安全组后，即可添加、更新或删除安全组的规则。在添加、更新或删除规则时，更改将自动应用于与安全组关联的资源。

所需的权限

在开始之前，请确保您拥有所需权限。有关更多信息，请参阅 管理安全组规则。

来源和目标

您可以将以下内容指定为入站规则的来源或出站规则的目标。

- 自定义 – IPv4 CIDR 块、IPv6 CIDR 块、其他安全组或前缀列表。
- Anywhere-IPv4 – 0.0.0.0/0 IPv4 CIDR 块。
- Anywhere-IPv6 – ::/0 IPv6 CIDR 块。
- 我的 IP – 本地计算机的公有 IPv4 地址。

> **Warning**
>
> 如果您选择 Anywhere-IPv4，则将允许来自所有 IPv4 地址的流量。如果您选择 Anywhere-IPv6，则将允许来自所有 IPv6 地址的流量。最佳做法是仅授权需要访问资源的特定 IP 地址范围。

使用控制台配置安全组规则

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择安全组。
3. 选择安全组。
4. 要编辑入站规则，请从操作或入站规则选项卡中选择编辑入站规则。
   a. 要添加规则，请选择添加规则，再输入规则的类型、协议、端口和来源。

      如果类型为 TCP 或 UDP，则必须输入允许的端口范围。对于自定义 ICMP，您必须从 Protocol（协议）中选择 ICMP 类型名称，并从 Port range（端口范围）中选择代码名称（如果适用）。对于任何其他类型，则会为您配置协议和端口范围。

b. 要更新规则，请根据需要更改规则的协议、描述和来源。但是，您无法更改来源类型。例如，若来源是 IPv4 CIDR 块，则无法指定 IPv6 CIDR 块、前缀列表或安全组。

c. 要删除规则，请选择规则的删除按钮。

5. 要编辑出站规则，请从操作或出站规则选项卡中选择编辑出站规则。

a. 要添加规则，请选择添加规则，再输入规则的类型、协议、端口和目标。您也可以输入可选描述。

如果类型为 TCP 或 UDP，则必须输入允许的端口范围。对于自定义 ICMP，您必须从 Protocol（协议）中选择 ICMP 类型名称，并从 Port range（端口范围）中选择代码名称（如果适用）。对于任何其他类型，则会为您配置协议和端口范围。

b. 要更新规则，请根据需要更改规则的协议、描述和来源。但是，您无法更改来源类型。例如，若来源是 IPv4 CIDR 块，则无法指定 IPv6 CIDR 块、前缀列表或安全组。

c. 要删除规则，请选择规则的删除按钮。

6. 选择保存规则。

使用 AWS CLI 配置安全组规则

- 添加 – 使用 authorize-security-group-ingress 和 authorize-security-group-egress 命令。
- 删除 – 使用 revoke-security-group-ingress 和 revoke-security-group-egress 命令。
- 修改 – 使用 modify-security-group-rules、update-security-group-rule-descriptions-ingress 和 update-security-group-rule-descriptions-egress 命令。

删除安全组

使用完创建的安全组后，可以将其删除。

要求

- 安全组无法与任何资源关联。
- 其他安全组中的规则无法引用安全组。
- 该安全组不能是 VPC 的默认安全组。

使用控制台删除安全组

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。

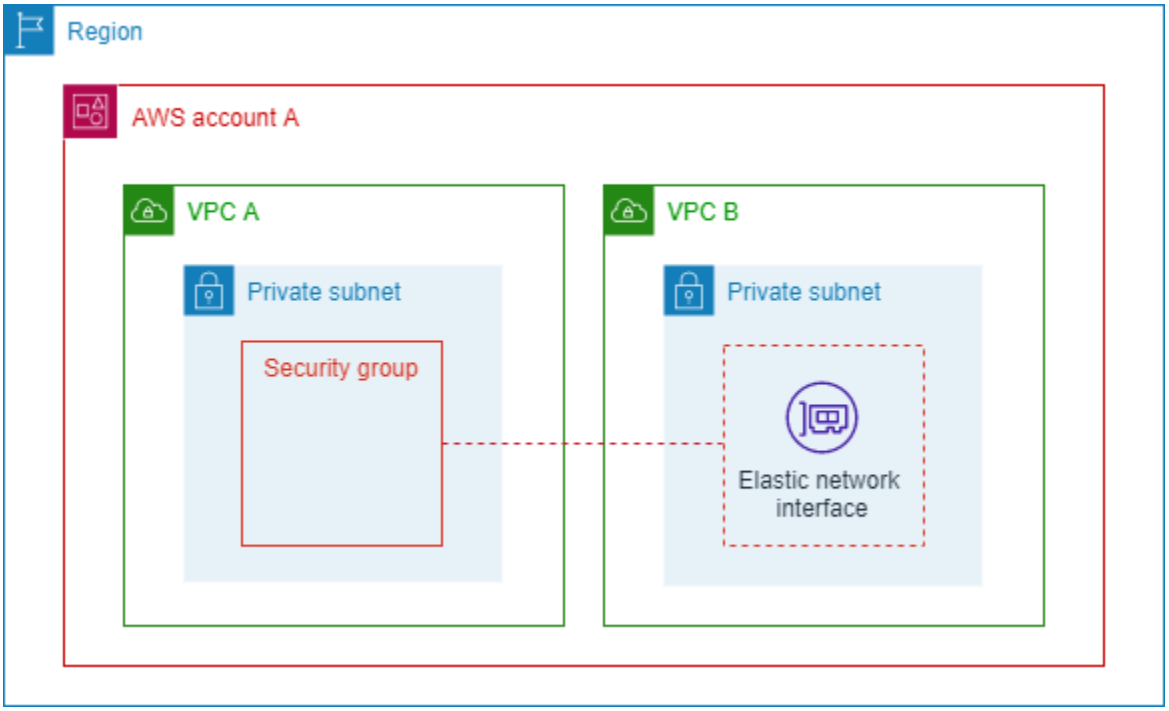
2. 在导航窗格中，选择安全组。
3. 选择安全组，然后依次选择操作、删除安全组。
4. 如果选择了多个安全组，系统会提示您进行确认。如果某些安全组无法删除，我们会显示每个安全组的状态，表明是否会被删除。要确认删除操作，请输入删除。
5. 选择删除。

使用 AWS CLI 删除安全组

使用 delete-security-group 命令。

将安全组与多个 VPC 关联

如果您的工作负载在具有共同网络安全要求的多个 VPC 中运行，则可以使用安全组 VPC 关联功能将安全组与同一区域中的多个 VPC 关联。这样，您就可以在一个位置管理和维护账户中多个 VPC 的安全组。



上图显示包含两个 VPC 的 AWS 账户 A。每个 VPC 都有在私有子网中运行的工作负载。在这种情况下，VPC A 和 B 子网中的工作负载具有相同的网络流量要求，因此账户 A 可以使用安全组 VPC 关联功能将 VPC A 中的安全组与 VPC B 关联。对关联安全组进行的任何更新都会自动应用于 VPC B 子网中工作负载的流量。

安全组 VPC 关联功能的要求

- 您必须拥有 VPC 或共享其中一个 VPC 子网才能将安全组与 VPC 关联。
- VPC 和安全组必须位于同一 AWS 区域。
- 您不能将默认安全组与其他 VPC 关联，也不能将安全组与默认 VPC 关联。
- 安全组所有者和 VPC 所有者都可以查看安全组 VPC 关联。

支持此功能的服务

- Amazon API Gateway（仅限 REST API）
- AWS Auto Scaling
- AWS CloudFormation
- Amazon EC2
- Amazon EFS
- Amazon EKS
- Amazon FSx
- AWS PrivateLink
- Amazon Route 53
- Elastic Load Balancing
  - 应用程序负载均衡器
  - 网络负载均衡器

将安全组与其他 VPC 关联

本节介绍如何使用 AWS Management Console和 AWS CLI 将安全组与 VPC 关联。

AWS Management Console

要将安全组与其他 VPC 关联

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在左侧导航窗格中，选择安全组。
3. 选择安全组以查看详细信息。

4. 选择 VPC 关联选项卡。
5. 选择 Associate VPC (关联 VPC)。
6. 在 VPC ID 下，选择要与安全组关联的 VPC。
7. 选择 Associate VPC (关联 VPC)。

Command line

要将安全组与其他 VPC 关联

1. 使用 associate-security-group-vpc 创建 VPC 关联。
2. 使用 describe-security-group-vpc-associations 查看 VPC 关联的状态，然后等待该状态变为 `associated`。

VPC 现已与安全组关联。

例如，将 VPC 与安全组关联后，您可以在 VPC 中启动一个实例并选择这一新的安全组，或者在现有安全组规则中引用此安全组。

取消安全组与其他 VPC 的关联

本节介绍如何使用 AWS Management Console和 AWS CLI 取消安全组与 VPC 的关联。如果您的目标是删除安全组，则可能需要这样做。如果安全组已关联，则无法将其删除。仅当关联的 VPC 中没有使用该安全组的网络接口时，您才能取消与该安全组的关联。

AWS Management Console

要取消安全组与 VPC 的关联

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在左侧导航窗格中，选择安全组。
3. 选择安全组以查看详细信息。
4. 选择 VPC 关联选项卡。
5. 选择取消关联 VPC。
6. 在 VPC ID 下，选择要与安全组取消关联的 VPC。
7. 选择取消关联 VPC。
8. 在 VPC 关联选项卡中查看取消关联的状态，然后等待该状态变为 `disassociated`。

Command line

要取消安全组与 VPC 的关联

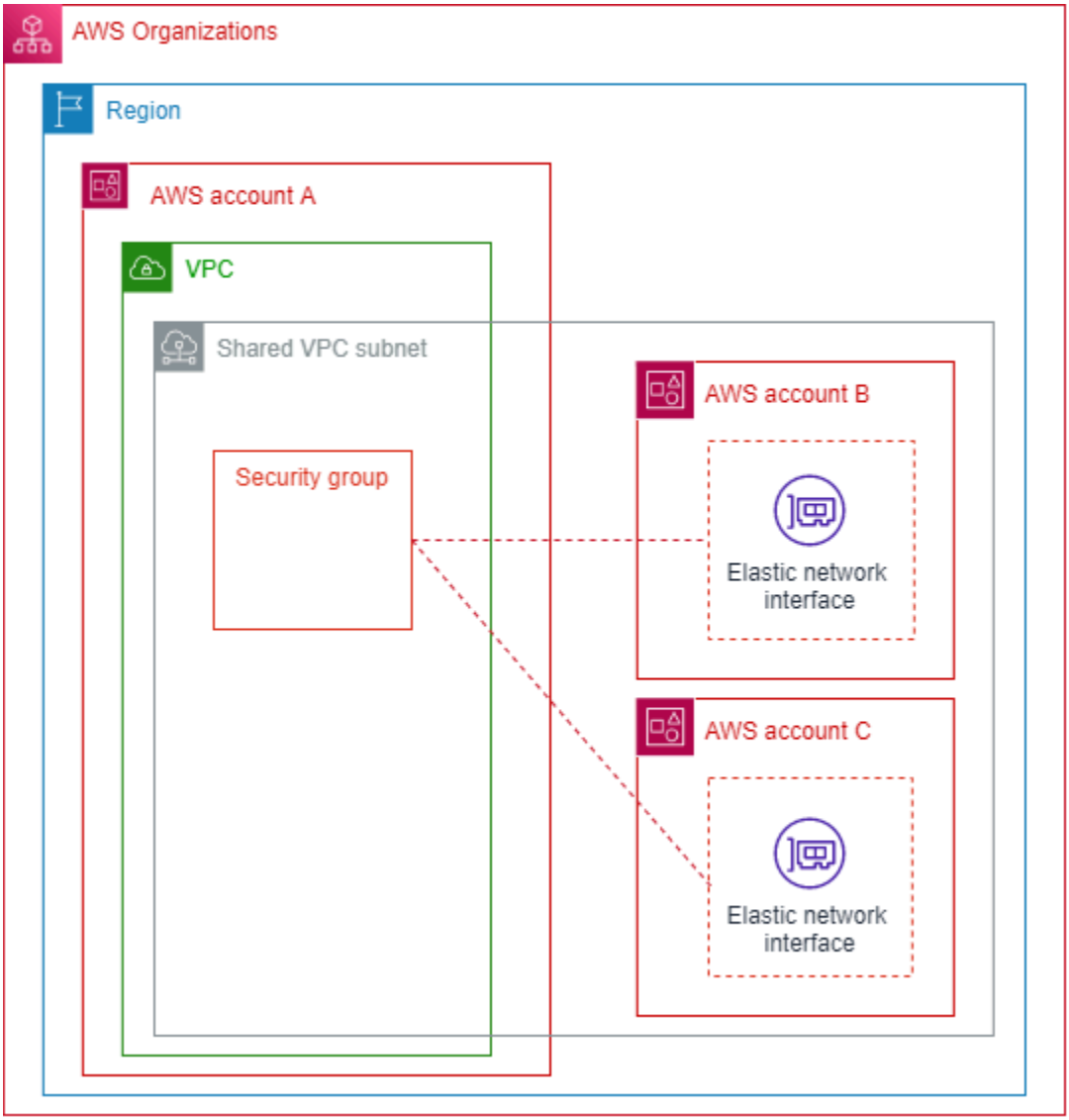
1. 使用 disassociate-security-group-vpc 取消与 VPC 的关联。
2. 使用 describe-security-group-vpc-associations 查看 VPC 取消关联的状态，然后等待该状态变为 `disassociated`。

VPC 现已取消与安全组的关联。

与 AWS Organizations 共享安全组

共享安全组功能使您能够与同一 AWS 区域中的其他 AWS Organizations 账户共享安全组，并使安全组可供这些账户使用。

下图演示如何使用共享安全组功能来简化 AWS Organizations 中的跨账户安全组管理：



此示意图显示属于同一组织的三个账户。账户 A 与账户 B 和账户 C 共享一个 VPC 子网。账户 A 使用共享安全组功能与账户 B 和账户 C 共享该安全组。然后，账户 B 和账户 C 在共享子网中启动实例时使用该安全组。这使账户 A 也能够管理该安全组；对安全组进行的任何更新都会应用于账户 B 和账户 C 在共享 VPC 子网中运行的资源。

共享安全组功能的要求

- 此功能仅适用于 AWS Organizations 中同一组织的账户。必须在 AWS Organizations 中启用资源共享。
- 共享安全组的账户必须同时拥有 VPC 和安全组。
- 您不能共享默认安全组。
- 您无法共享位于默认 VPC 中的安全组。

- 参与者账户可以在共享 VPC 中创建安全组，但无法共享这些安全组。
- IAM 主体需要一组最低权限才能与 AWS RAM 共享一个安全组。使用 `AmazonEC2FullAccess` 和 `AWSResourceAccessManagerFullAccess` 托管 IAM 策略，确保 IAM 主体拥有共享和使用所共享安全组所需的权限。如果使用自定义 IAM 策略，则需要进行 `c2:PutResourcePolicy` 和 `ec2:DeleteResourcePolicy` 操作。这些是仅限权限的 IAM 操作。如果 IAM 主体未获得这些权限，则在尝试使用 AWS RAM 共享安全组时将出错。

支持此功能的服务

- Amazon API Gateway
- Amazon EC2
- Amazon ECS
- Amazon EFS
- Amazon EKS
- Amazon EMR
- Amazon FSx
- Amazon ElastiCache
- AWS Elastic Beanstalk
- AWS Glue
- Amazon MQ
- Amazon SageMaker AI
- Elastic Load Balancing
  - 应用程序负载均衡器
  - 网络负载均衡器

此功能如何影响现有的配额

安全组配额适用。但是，对于“每个网络接口的安全组数”配额，如果参与者在弹性网络接口（ENI）上同时使用自有和共享的组，则所有者和参与者的最低配额适用。

演示此功能如何影响配额的示例：

- 所有者账户配额：每个接口 4 个安全组
- 参与者账户配额：每个接口 5 个安全组。

- 所有者与参与者共享的组 SG-O1、SG-O2、SG-O3、SG-O4、SG-O5。参与者在 VPC 中已经有自己的组：SG-P1、SG-P2、SG-P3、SG-P4、SG-P5。
- 如果参与者创建 ENI 并且仅使用自己的组，则他们可以关联所有 5 个安全组（SG-P1、SG-P2、SG-P3、SG-P4、SG-P5），因为这是他们的配额。
- 如果参与者创建 ENI 并在其上使用任何共享的组，则他们最多只能关联 4 个组。在这种情况下，此类 ENI 的配额是所有者和参与者的最低配额。可能的有效配置将如下所示：
  - SG-O1、SG-P1、SG-P2、SG-P3
  - SG-O1、SG-O2、SG-O3、SG-O4

共享安全组

本节介绍如何使用 AWS Management Console和 AWS CLI 与组织中的其他账户共享安全组。

AWS Management Console

要共享安全组

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在左侧导航窗格中，选择安全组。
3. 选择安全组以查看详细信息。
4. 选择 Sharing (共享) 选项卡。
5. 选择共享安全组。
6. 选择创建资源共享。这样，AWS RAM 控制台就会打开，您将在其中为安全组创建资源共享。
7. 输入资源共享的名称。
8. 在资源 – 可选下，选择安全组。
9. 选择安全组。该安全组不能是默认安全组，也不能与默认 VPC 关联。
10. 选择下一步。
11. 查看允许主体执行的操作，然后选择下一步。
12. 在主体 – 可选下，选择仅允许在企业内共享。
13. 在主体下，选择以下主体类型之一，然后输入相应的数字：
    - AWS 账户：您的组织中账户的账号。
    - 组织：AWS Organizations ID。
    - 组织单位（OU）：组织中 OU 的 ID。

- IAM 角色：IAM 角色的 ARN。创建该角色的账户必须与创建此资源共享的账户属于同一组织。
- IAM 用户：IAM 用户的 ARN。创建该用户的账户必须与创建此资源共享的账户属于同一组织。
- 服务主体：您不能与服务主体共享安全组。

14. 选择添加。
15. 选择下一步。
16. 选择创建资源共享。
17. 在共享资源下，等待看到状态为 `Associated`。如果安全组关联失败，则可能是由于上面列出的限制之一所致。查看安全组的详细信息以及详细信息页面上的共享选项卡，以查看与安全组不可共享的原因相关的所有消息。
18. 返回 VPC 控制台安全组列表。
19. 选择您共享的安全组。
20. 选择 Sharing (共享) 选项卡。您的 AWS RAM 资源应该在其中显示。如果未显示，则资源共享创建可能失败，您可能需要重新创建。

Command line

要共享安全组

1. 您必须先为想要与 AWS RAM 共享的安全组创建资源共享。有关如何使用 AWS CLI 创建与 AWS RAM 的资源共享的步骤，请参阅《AWS RAM User Guide》中的 Creating a resource share in AWS RAM
2. 要查看创建的资源共享关联，请使用 get-resource-share-associations。

安全组现已共享。您可以在同一 VPC 内的共享子网中启动 EC2 实例时选择安全组。

停止共享安全组

本节介绍如何使用 AWS Management Console和 AWS CLI 停止与组织中的其他账户共享安全组。

AWS Management Console

要停止共享安全组

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。

2. 在左侧导航窗格中，选择安全组。
3. 选择安全组以查看详细信息。
4. 选择 Sharing (共享) 选项卡。
5. 选择安全组资源共享，然后选择停止共享。
6. 选择是的，停止共享。

Command line

要停止共享安全组

使用 delete-resource-share 删除资源共享。

该安全组不再共享。在所有者停止共享安全组后，以下规则适用：

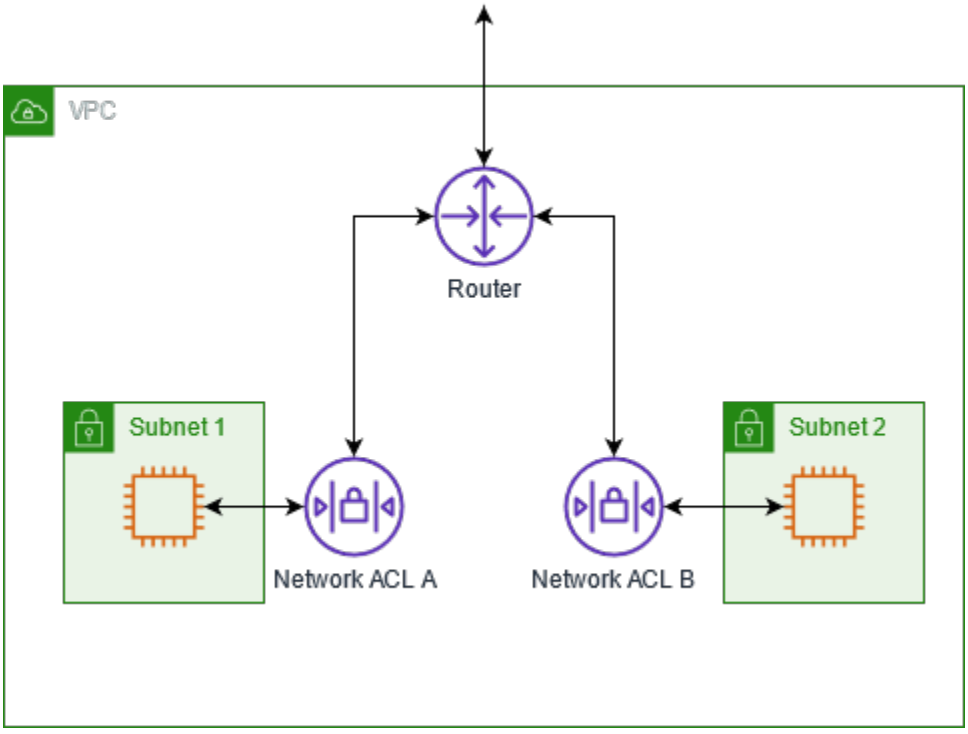
- 现有的参与者弹性网络接口（ENI）将继续获取对已取消共享的安全组进行的任何安全组规则更新。取消共享只会阻止参与者与已取消共享组创建新关联。
- 参与者无法再将已取消共享的安全组与其拥有的任何 ENI 关联。
- 参与者可以描述和删除仍与已取消共享安全组关联的 ENI。
- 如果参与者仍具有与已取消共享的安全组关联的 ENI，则所有者无法删除已取消共享的安全组。仅当参与者从其所有 ENI 取消关联（删除）安全组后，所有者才能删除该安全组。
- 参与者不能使用与非共享安全组关联的 ENI 启动新的 EC2 实例。

使用网络访问控制列表控制子网流量

网络访问控制列表 (ACL) 在子网级别允许或拒绝特定的入站或出站流量。您可以使用 VPC 的默认网络 ACL，也可以为 VPC 创建自定义网络 ACL，使其规则与您安全组的规则相似，以便为您的 VPC 添加额外安全层。

使用网络 ACL 不会产生任何额外的费用。

以下示意图显示了具有两个子网的 VPC。每个子网都有网络 ACL。当流量（例如，来自对等 VPC、VPN 连接或互联网）进入 VPC 时，路由器会将流量发送到其目的地。网络 ACL A 确定哪些发往子网 1 的流量可以进入子网 1，哪些发往子网 1 位置以外的流量可以离开子网 1。同样，网络 ACL B 确定哪些流量可以进入和离开子网 2。



有关安全组和网络 ACL 之间区别的更多信息，请参阅 比较安全组和网络 ACL。

内容

- 网络 ACL 基础知识
- 网络 ACL 规则
- VPC 的默认网络 ACL
- 您的 VPC 的自定义网络 ACL
- 路径 MTU 发现和网络 ACL
- 为您的 VPC 创建网络 ACL
- 管理 VPC 的网络 ACL 关联
- 删除 VPC 的网络 ACL
- 示例：控制对子网中的实例的访问

网络 ACL 基础知识

以下是开始之前需要了解的有关网络 ACL 的基本信息。

网络 ACL 关联

- 您的 VPC 中的每个子网都必须与一个网络 ACL 相关联。如果您没有明确地将子网与网络 ACL 关联，则子网将自动与默认网络 ACL 关联。
- 您可以创建自定义网络 ACL 并将其与子网相关联，以允许或拒绝子网级别的特定入站或出站流量。
- 您可以将网络 ACL 与多个子网关联。但是，一个子网一次只能与一个网络 ACL 关联。当您将一个网络 ACL 与一个子网关联时，将删除之前的关联。

网络 ACL 规则

- 网络 ACL 具有入站规则和出站规则。每个网络 ACL 的规则数量有配额（或限制）。每条规则都可以接受或拒绝流量。每条规则都有介于 1 到 32766 之间的数字。在决定是否接受或拒绝流量时，我们按顺序评估规则，从编号最低的规则开始。如果流量与规则匹配，则应用该规则，并且我们不会评估其他任何规则。我们建议您首先以增量方式创建规则（例如，以 10 或 100 的增量增加），以便日后需要时插入新的规则。
- 在流量进入和离开子网时，我们会评估网络 ACL 规则，而不是在子网内路由流量时进行评估。
- NACL 无状态，这意味着不会保存有关先前发送或接收的流量的信息。例如，如果您创建了 NACL 规则，允许流向子网的特定入站流量，则不会自动允许对该流量做出响应。这与安全组的工作原理截然不同。安全组有状态，这意味着会保存有关先前发送或接收的流量的信息。例如，如果安全组允许流向 EC2 实例的入站流量，则将自动允许响应，不受任何出站安全组规则的影响。

限制

- 网络 ACL 有配额（也称为限制）。有关更多信息，请参阅 网络 ACL。
- 网络 ACL 无法阻止发往或来自 Route 53 Resolver 的 DNS 请求（也称为 VPC+2 IP 地址或 AmazonProvidedDNS）。要通过 Route 53 Resolver 筛选 DNS 请求，您可以启用 Route 53 Resolver DNS 防火墙。
- 网络 ACL 无法阻止访问实例元数据服务（IMDS）的流量。要管理对 IMDS 的访问权限，请参阅《Amazon EC2 用户指南》中的 配置实例元数据选项。
- 网络 ACL 不会筛选发往和来自以下位置的流量：
  - Amazon 域名服务 (DNS)
  - Amazon 动态主机配置协议 (DHCP)
  - Amazon EC2 实例元数据
  - Amazon ECS 任务元数据端点

- Windows 实例的许可证激活
- Amazon Time Sync Service
- 默认 VPC 路由器使用的预留 IP 地址

网络 ACL 规则

您可以在默认网络 ACL 中添加或删除规则，或为您的 VPC 创建额外网络 ACL。当您在网络 ACL 中添加或删除规则时，更改也会自动应用到与其相关联的子网。

以下为部分网络 ACL 规则：

- 规则编号。规则评估从编号最低的规则起开始进行。只要有一条规则与流量匹配，即应用该规则，并忽略与之冲突的任意更大编号的规则。
- 类型。流量的类型，例如 SSH。您也可以指定所有流量或自定义范围。
- 协议。您可以指定任何有标准协议编号的协议。有关更多信息，请参阅 Protocol Numbers。如果您指定 ICMP 作为协议，您可以指定任意或全部 ICMP 类型和代码。
- 端口范围。流量的侦听端口或端口范围。例如，80 用于 HTTP 流量。
- 源。[仅限入站规则]流量的源（CIDR 范围）。
- 目的地。[仅限出站规则]流量的目的地（CIDR 范围）。
- 允许/拒绝。允许还是拒绝指定的流量。

有关规则示例，请参阅 the section called “示例：控制对子网中的实例的访问”。

注意事项

- 每个网络 ACL 的规则数量存在配额（也称为限制）。有关更多信息，请参阅 Amazon VPC 配额。
- 当您在网络 ACL 中添加或删除规则时，与其相关联的子网也会随之更改。这些更改在短时间内生效。
- 如果您使用命令行工具或 Amazon EC2 API 添加规则，则系统会自动将 CIDR 范围修改为其规范形式。例如，如果您为 CIDR 范围指定 `100.68.0.18/18`，我们将创建一个 CIDR 范围为 `100.68.0.0/18` 的规则。
- 在您必须开放一系列端口、同时在此部分端口内您想拒绝部分端口，您可能希望添加一项拒绝规则。请务必确保拒绝规则的给定数量要小于允许更大范围端口流量的规则的数量。

- 如果您同时在网络 ACL 中添加和删除规则，则请谨慎操作。如果您删除入站或出站规则，然后添加的新条目数超过了允许的条目数（请参阅 Amazon VPC 配额），则将移除已选择删除的条目，且不会添加新条目。这可能会导致意外的连接问题，并阻止进出 VPC 的访问。

VPC 的默认网络 ACL

虚拟私有云（VPC）自动带有默认网络 ACL。默认网络 ACL 配置为让所有流量流进和流出与其关联的子网。每个网络 ACL 还包含以星号（*）为规则编号的规则。这些规则确保在数据包与任何其他编号规则不匹配时拒绝该数据包。

您可以通过添加规则或删除默认编号规则，来修改默认网络 ACL。您无法删除规则编号是星号的规则。

默认入站规则

下表显示默认网络 ACL 的默认入站规则。仅当您创建具有关联 IPv6 CIDR 块的 VPC 或将 IPv6 CIDR 块与 VPC 关联时，才会添加 IPv6 的规则。但是，如果您修改了默认网络 ACL 的入站规则，在将 IPv6 块与 VPC 关联时，我们不会添加允许所有入站 IPv6 流量的规则。

规则 #	类型	协议	端口范围	源	允许/拒绝
100	所有 IPv4 流量	全部	全部	0.0.0.0/0	允许
101	所有 IPv6 流量	All	All	::/0	允许
*	所有流量	All	全部	0.0.0.0/0	DENY
*	所有 IPv6 流量	All	All	::/0	DENY

默认出站规则

下表显示默认网络 ACL 的默认出站规则。仅当您创建具有关联 IPv6 CIDR 块的 VPC 或将 IPv6 CIDR 块与 VPC 关联时，才会添加 IPv6 的规则。但是，如果您修改了默认网络 ACL 的出站规则，在将 IPv6 块与 VPC 关联时，我们不会添加允许所有出站 IPv6 流量的规则。

规则 #	类型	协议	端口范围	目的地	允许/拒绝
100	所有流量	All	全部	0.0.0.0/0	允许
101	所有 IPv6 流量	All	All	::/0	允许
*	所有流量	All	全部	0.0.0.0/0	DENY
*	所有 IPv6 流量	All	All	::/0	DENY

您的 VPC 的自定义网络 ACL

您可以创建自定义网络 ACL 并将其与子网相关联，以允许或拒绝子网级别的特定入站或出站流量。有关更多信息，请参阅 the section called “创建网络 ACL”。

每个网络 ACL 都包含一个默认入站规则和一个默认出站规则，其规则编号是星号（*）。这些规则确保在数据包与任何其他规则不匹配时拒绝该数据包。

您可以通过添加或删除规则，来修改网络 ACL。您无法删除规则编号是星号的规则。

对于您添加的每个规则，都必须存在允许响应流量的入站或出站规则。有关如何选择适当的临时端口的更多信息，请参见 临时端口。

示例入站规则

下表显示网络 ACL 的示例入站规则。仅当 VPC 具有关联的 IPv6 CIDR 块时，才添加 IPv6 的规则。IPv4 和 IPv6 流量是单独评估的。因此，所有 IPv4 流量的规则都不适用于 IPv6 流量。您可以在相应 IPv4 规则旁边添加 IPv6 规则，也可以在最后一个 IPv4 规则之后添加 IPv6 规则。

随着数据包流向子网，我们会根据与子网关联的网络 ACL 的入站规则评估数据包（从编号最低的规则开始）。例如，假设存在发往 HTTPS 端口 (443) 的 IPv4 流量。数据包与规则 100 或 105 不匹配。它与规则 110 匹配，后者允许流量进入子网。如果数据包已发往端口 139 (NetBIOS)，则它与任何编号规则均不匹配，因此 IPv4 流量的 * 规则最终会拒绝此数据包。

规则 #	类型	协议	端口范围	源	允许/拒绝	注释
100	HTTP	TCP	80	0.0.0.0/0	允许	允许来自任意 IPv4 地址的入站 HTTP 流量。
105	HTTP	TCP	80	::/0	允许	允许来自任意 IPv6 地址的入站 HTTP 流量。
110	HTTPS	TCP	443	0.0.0.0/0	允许	允许来自任意 IPv4 地址的入站 HTTPS 流量。
115	HTTPS	TCP	443	::/0	允许	允许来自任意 IPv6 地址的入站 HTTPS 流量。
120	SSH	TCP	22	*192.0.2.0 /24*	允许	允许来自您的家庭网络的公有 IPv4 地址范围的入站 SSH 流量（通过互联网网关）。
140	自定义 TCP	TCP	*32768-65535*	0.0.0.0/0	允许	允许来自互联网的入站返回 IPv4 流量（对于源自子网的请求）。
145	自定义 TCP	TCP	*32768-65535*	::/0	允许	允许来自互联网的入站返回 IPv6 流量（对于源自子网的请求）。
*	所有流量	All	全部	0.0.0.0/0	拒绝	拒绝所有未经前置规则（不可修改）处理的入站 IPv4 流量。

规则 #	类型	协议	端口范围	源	允许/拒绝	注释
*	所有流量	All	All	::/0	拒绝	拒绝所有未经前置规则（不可修改）处理的入站 IPv6 流量。

示例出站规则

下表显示自定义网络 ACL 的示例出站规则。仅当 VPC 具有关联的 IPv6 CIDR 块时，才添加 IPv6 的规则。IPv4 和 IPv6 流量是单独评估的。因此，所有 IPv4 流量的规则都不适用于 IPv6 流量。您可以在相应 IPv4 规则旁边添加 IPv6 规则，也可以在最后一个 IPv4 规则之后添加 IPv6 规则。

规则 #	类型	协议	端口范围	目的地	允许/拒绝	注释
100	HTTP	TCP	80	0.0.0.0/0	允许	允许出站 IPv4 HTTP 流量从子网流向 Internet。
105	HTTP	TCP	80	::/0	允许	允许出站 IPv6 HTTP 流量从子网流向 Internet。
110	HTTPS	TCP	443	0.0.0.0/0	允许	允许出站 IPv4 HTTPS 流量从子网流向 Internet。
115	HTTPS	TCP	443	::/0	允许	允许出站 IPv6 HTTPS 流量从子网流向 Internet。
120	自定义 TCP	TCP	*1024-655 5*	*192.0.2.(/24*	允许	允许从您的家庭网络发出对 SSH 流量的出站响应。
140	自定义 TCP	TCP	*32768-65 35*	0.0.0.0/0	允许	允许对互联网客户端的出站 IPv4 响应（例如，提供网页）。

规则 #	类型	协议	端口范围	目的地	允许/拒绝	注释
145	自定义 TCP	TCP	32768-65535	::/0	允许	允许对互联网客户端的出站 IPv6 响应（例如，提供网页）。
*	所有流量	All	全部	0.0.0.0/0	DENY	拒绝所有未经前置规则处理的出站 IPv4 流量。
*	所有流量	All	All	::/0	DENY	拒绝所有未经前置规则处理的出站 IPv6 流量。

临时端口

上一个部分中的网络 ACL 实例使用了临时端口范围 32768-65535。但是，您可能需要根据自己使用的或作为通信目标的客户端的类型为网络 ACL 使用不同的范围。

发起请求的客户端会选择临时端口范围。根据客户端的操作系统不同，范围也随之更改。

- 许多 Linux 内核（包括 Amazon Linux 内核）使用端口 32768-61000。
- 源自 Elastic Load Balancing 的请求使用端口 1024-65535。
- Windows 操作系统通过 Windows Server 2003 使用端口 1025-5000。
- Windows Server 2008 及更高版本使用端口 49152-65535。
- NAT 网关使用端口 1024 - 65535。
- AWS Lambda 函数使用端口 1024-65535。

例如，如果一个来自 Internet 上的 Windows 10 客户端的请求到达您的 VPC 中的 Web 服务器，则您的网络 ACL 必须有相应的出站规则，才能支持目标为端口 49152-65535 的流量。

如果您的 VPC 中的一个实例是发起请求的客户端，则您的网络 ACL 必须有入站规则来支持发往特定于实例的操作系统的临时端口的流量。

在实际中，为使不同客户端类型可以启动流量进入您 VPC 中的公有实例，您可以开放临时端口 1024-65535。但是，您也可以在 ACL 中添加规则以拒绝任何在此范围内的来自恶意端口的数据流。请务必将拒绝规则放在表的较前端，先于开放一系列临时端口的允许规则。

自定义网络 ACL 和其他 AWS 服务

如果您创建自定义网络 ACL，请注意它可能会如何影响您使用其他 AWS 服务创建的资源。

借助 Elastic Load Balancing，如果您的后端实例的子网有一个网络 ACL，并且您在其中针对源为 `0.0.0.0/0` 或子网的 CIDR 的所有流量添加了拒绝规则，则您的负载均衡器将无法对这些实例执行运行状况检查。有关负载均衡器和后端实例的推荐网络 ACL 规则的更多信息，请参阅以下内容：

- Network ACLs for your Application Load Balancer
- Network ACLs for your Network Load Balancer
- Network ACLs for your Classic Load Balancer

排查可达性问题

Reachability Analyzer 是一款静态配置分析工具。使用 Reachability Analyzer 可分析和调试 VPC 中两个资源之间的网络可达性。如果可以访问这些资源，则 Reachability Analyzer 会生成有关这些资源间虚拟路径的逐跳详细信息，否则会确定障碍组件。例如，其可以识别缺失或配置错误的网络 ACL 规则。

有关更多信息，请参阅 Reachability Analyzer 角色指南。

路径 MTU 发现和网络 ACL

路径 MTU 发现用于确定两台设备之间的路径 MTU。路径 MTU 是原始主机和接收主机之间的路径所支持的最大数据包大小。

对于 IPv4，如果主机发送一个大于接收主机的 MTU 或大于路径上某台设备的 MTU 的数据包，则接收主机或设备将删除此数据包，然后返回以下 ICMP 消息：`Destination Unreachable: Fragmentation Needed and Don't Fragment was Set`（类型 3，代码 4）。这将指示传输主机将有效负载拆分为多个较小的数据包，然后重新传输。

IPv6 协议不支持网络中的分段。如果主机发送一个大于接收主机的 MTU 或大于路径上某台设备的 MTU 的数据包，则接收主机或设备将删除此数据包，然后返回以下 ICMP 消息：`ICMPv6 Packet Too Big (PTB)`（类型 2）。这将指示传输主机将有效负载拆分为多个较小的数据包，然后重新传输。

如果您子网中主机之间的最大传输单位 (MTU) 不同，或您的实例可以通过互联网与对等项通信，则必须添加以下网络 ACL 规则（入站和出站）。这可确保路径 MTU 发现能够正常工作并防止数据包丢失。为类型选择自定义 ICMP 规则，为端口范围选择目的地无法到达、需要分段，DF 标志已设置（类型 3，代码 4）。如果您使用 traceroute，还需添加以下规则：选择自定义 ICMP 规则作为类型，并选

择超时和 TTL 中转过期作为端口范围（类型 11，代码 0）。有关更多信息，请参阅《Amazon EC2 用户指南》中的 EC2 实例的网络最大传输单位（MTU）。

为您的 VPC 创建网络 ACL

以下任务为您展示如何创建网络 ACL，向网络 ACL 添加规则，然后将网络 ACL 与子网关联。

任务

- 步骤 1：创建网络 ACL
- 步骤 2：添加规则
- 步骤 3：将子网与网络 ACL 关联
- （可选）使用 Firewall Manager 管理网络 ACL

步骤 1：创建网络 ACL

您可以为 VPC 创建自定义网络 ACL。自定义网络 ACL 的初始规则会阻止所有入站和出站流量。默认情况下，您的新自定义网络 ACL 未与子网关联，必须与子网显式关联。

要使用控制台创建网络 ACL

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Network ACLs（网络 ACL）。
3. 选择创建网络 ACL。
4. （可选）对于名称，输入网络 ACL 的名称。
5. 对于 VPC，选择 VPC。
6. （可选）对于标签，选择添加标签，然后输入标签键和标签值。
7. 选择创建网络 ACL。

要使用命令行创建网络 ACL

- create-network-acl (AWS CLI)
- New-EC2NetworkAcl (AWS Tools for Windows PowerShell)

步骤 2：添加规则

您可以添加允许或拒绝入站或出站流量的规则。

我们会按顺序处理规则，以编号最低的规则开始。我们建议您使用跳跃的规则编号（例如 100、200、300）而不是使用顺序编号（例如 101、102、103）。这会让添加新规则变得更加简单，无需对现有规则重新编号。

如果您使用 Amazon EC2 API 或命令行工具，则无法修改规则。您只能添加和删除规则。如果您使用 Amazon VPC 控制台，则可以修改现有规则的条目。控制台将为您删除现有规则并添加新规则。如果您需要更改 ACL 中的规则顺序，您必须添加有新规则编号的新规则，并随后删除最初的规则。

要使用控制台向网络 ACL 添加规则

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Network ACLs（网络 ACL）。
3. 选择网络 ACL。
4. 要添加入站规则，请执行以下操作：
   a. 选择入站规则选项卡。
   b. 选择编辑入站规则和添加新规则。
   c. 输入尚未使用的规则编号、类型、协议、端口范围、来源，以及是允许还是拒绝流量。对于某些类型，我们会在协议和端口中为您提供。如果系统提示您提供端口范围，则请输入端口号或端口范围（例如 49152-65535）。

      要使用某个未列出的协议，请选择自定义协议作为类型，然后选择该协议。有关更多信息，请参阅 IANA 协议编号。
   d. 选择保存更改。
5. 要添加出站规则，请执行以下操作：
   a. 选择 Outbound rules（出站规则）选项卡。
   b. 选择编辑出站规则和添加新规则。
   c. 输入尚未使用的规则编号、类型、协议、端口范围、来源，以及是允许还是拒绝流量。对于某些类型，我们会在协议和端口中为您提供。如果系统提示您提供端口范围，则请输入端口号或端口范围（例如 49152-65535）。

      要使用某个未列出的协议，请选择自定义协议作为类型，然后选择该协议。有关更多信息，请参阅 IANA 协议编号。
   d. 选择保存更改。

要使用命令行向网络 ACL 添加规则

- create-network-acl-entry (AWS CLI)
- New-EC2NetworkAclEntry (AWS Tools for Windows PowerShell)

要使用命令行替换网络 ACL 中的规则

- replace-network-acl-entry (AWS CLI)
- Set-EC2NetworkAclEntry (AWS Tools for Windows PowerShell)

要使用命令行从网络 ACL 中删除规则

- delete-network-acl-entry (AWS CLI)
- Remove-EC2NetworkAclEntry (AWS Tools for Windows PowerShell)

步骤 3：将子网与网络 ACL 关联

如需对特定子网应用特定的网络 ACL 规则，您必须首先将子网与网络 ACL 关联。您可以将网络 ACL 与多个子网关联。但是，一个子网只能与一个网络 ACL 关联。任何未与特定 ACL 关联的子网都会默认与默认网络 ACL 关联。

将子网与网络 ACL 关联

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在导航窗格中，选择 Network ACLs（网络 ACL），然后选择网络 ACL。
3. 在详细信息窗格中的 Subnet Associations（子网关联）选项卡上，选择 Edit（编辑）。选中要与网络 ACL 关联的子网的 Associate（关联）复选框，然后选择 Save（保存）。

（可选）使用 Firewall Manager 管理网络 ACL

AWS Firewall Manager 可简化跨多个账户和子网的网络 ACL 管理和维护任务。您可以使用 Firewall Manager 监视组织中的账户和子网，并自动应用您定义的网络 ACL 配置。如果要保护整个企业，或者经常添加要通过中央管理员账户自动保护的新子网，Firewall Manager 尤其有用。

利用 Firewall Manager 网络 ACL 策略，借助单个管理员账户，您可以配置、监视和管理您希望在整个组织中使用的网络 ACL 中定义的最小规则集。您可以指定组织的哪些账户和子网在 Firewall Manager

策略的范围内。Firewall Manager 报告范围内子网的网络 ACL 的合规性状态，您可以将 Firewall Manager 配置为自动修复不合规的网络 ACL。

有关更多信息，请参阅《AWS Firewall Manager 开发人员指南》中的以下资源：

- AWS Firewall Manager 先决条件
- 设置 AWS Firewall Manager 网络 ACL 策略
- 将网络 ACL 策略与 Firewall Manager 搭配使用

管理 VPC 的网络 ACL 关联

每个子网都与一个网络 ACL 关联。当您首次创建子网时，它会与 VPC 的默认网络 ACL 关联。您可以创建自定义网络 ACL 并将其与一个或多个子网关联，来替换之前的网络 ACL 关联。

任务

- 描述您的网络 ACL 关联
- 更改与网络 ACL 关联的子网
- 更改与子网关联的网络 ACL

描述您的网络 ACL 关联

您可以描述与子网关联的网络 ACL，也可以描述与网络 ACL 关联的子网。

要描述使用控制台与子网关联的网络 ACL

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Subnets（子网）。
3. 选择子网。
4. 选择网络 ACL 选项卡。

要使用 AWS CLI 描述与子网关联的网络 ACL

使用以下 describe-network-acls 命令列出与指定子网关联的网络 ACL。

```
aws ec2 describe-network-acls --filters Name=association.subnet-
id,Values=subnet-0d2d1b81e0bc9c6d4 --query NetworkAcls[*].NetworkAclId
```

下面是示例输出。

```
[
    "acl-03701d1f82d8c3fd6"
]
```

要使用控制台描述与网络 ACL 关联的子网

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Network ACLs（网络 ACL）。
3. 选择网络 ACL。
4. 选择子网关联选项卡。

要使用 AWS CLI 描述与网络 ACL 关联的子网

使用以下 describe-network-acls 命令列出与指定网络 ACL 关联的子网。

```
aws ec2 describe-network-acls --network-acl-ids acl-060415a18fcc9afde --query
 NetworkAcls[*].Associations[].SubnetId
```

下面是示例输出。

```
[
    "subnet-0d2d1b81e0bc9c6d4",
    "subnet-0e990c67809773b19",
    "subnet-0eb17d85f5dfd33b1",
    "subnet-0e01d500780bb7468"
]
```

更改与网络 ACL 关联的子网

您可以从子网取消自定义网络 ACL 的关联。在您将子网与自定义网络 ACL 取消关联后，我们会自动将其与 VPC 的默认网络 ACL 关联。这些更改在短时间内生效。

要更改与网络 ACL 关联的子网

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Network ACLs（网络 ACL）。
3. 选择网络 ACL。

4. 选择操作和编辑子网关联。
5. 从选定子网中移除子网。
6. 选择保存更改。

更改与子网关联的网络 ACL

您可以更改与某个子网关联的网络 ACL。例如，当您创建一个子网时，该子网会最初与 VPC 的默认网络 ACL 关联。如果创建自定义网络 ACL，则通过将网络 ACL 与一个或多个子网关联来应用网络 ACL 规则。

更改子网的网络 ACL 后，更改将在短时间内生效。

要更改与子网关联的网络 ACL

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Subnets（子网）。
3. 选择子网。
4. 选择操作和编辑网络 ACL 关联。
5. 对于网络 ACL ID，选择要与子网关联的网络 ACL，并查看所选网络 ACL 的入站和出站规则。
6. 选择保存。

要使用命令行替换网络 ACL 关联

- replace-network-acl-association (AWS CLI)
- Set-EC2NetworkAclAssociation (AWS Tools for Windows PowerShell)

删除 VPC 的网络 ACL

当不再使用某个网络 ACL 时，可以将其删除。如果网络 ACL 有关联子网，则无法删除该网络 ACL。您无法删除默认网络 ACL。

要使用控制台从网络 ACL 中移除子网关联

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Network ACLs（网络 ACL）。关联对象列指示与每个网络 ACL 关联的子网数。如果没有关联的子网，则此列为 -。

3. 选择网络 ACL。
4. 选择操作和编辑子网关联。
5. 移除子网关联。
6. 选择保存更改。

要使用命令行描述您的网络 ACL，包括关联

- describe-network-acls (AWS CLI)
- Get-EC2NetworkAcl (AWS Tools for Windows PowerShell)

要使用命令行替换网络 ACL 关联

- replace-network-acl-association (AWS CLI)
- Set-EC2NetworkAclAssociation (AWS Tools for Windows PowerShell)

要使用控制台删除网络 ACL

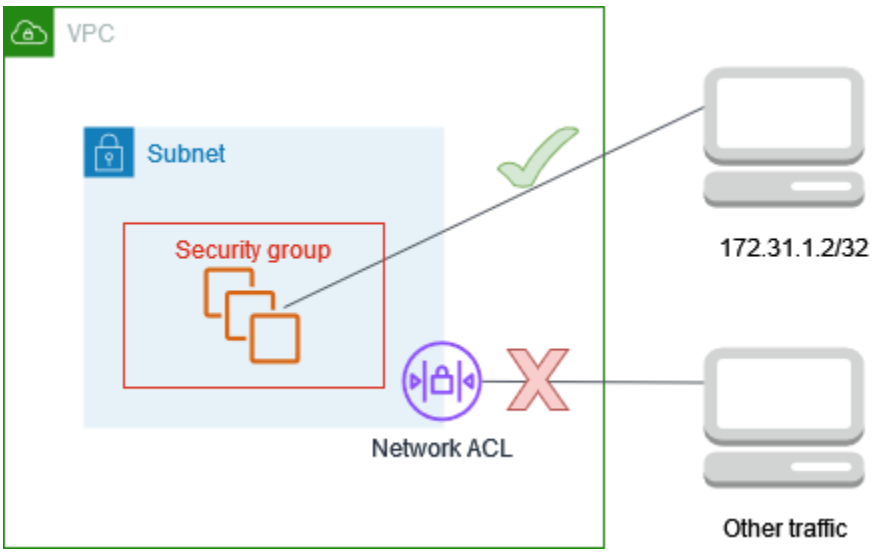
1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在导航窗格中，选择 Network ACLs（网络 ACL）。
3. 选择网络 ACL。
4. 选择操作和删除网络 ACL。
5. 提示进行确认时，输入 **`delete`**，然后选择 Delete（删除）。

要使用命令行删除网络 ACL

- delete-network-acl (AWS CLI)
- Remove-EC2NetworkAcl (AWS Tools for Windows PowerShell)

示例：控制对子网中的实例的访问

在本示例中，子网中的任意两个实例可相互通信，并可从受信任的远程计算机访问它们，以执行管理任务。远程计算机可能是本地网络中的计算机（如图所示），也可能是其他子网或 VPC 中的实例。子网的网络 ACL 规则和实例的安全组规则允许从远程计算机的 IP 地址进行访问。来自 Internet 或其他网络的所有其他流量会被拒绝。



使用网络 ACL 让您能够灵活地更改实例的安全组或安全组规则，同时依赖网络 ACL 作为备份防御层。例如，如果意外更新安全组以允许从任何地方进行入站 SSH 访问，但网络 ACL 仅允许从远程计算机的 IP 地址范围访问，则网络 ACL 会拒绝来自任何其他 IP 地址的入站 SSH 流量。

网络 ACL 规则

下面是与子网关联的网络 ACL 的示例入站规则。这些规则应用到子网中的所有实例。

规则 #	类型	协议	端口范围	源	允许/拒绝	注释
100	SSH	TCP	22	*172.31.1.2/32*	允许	允许远程计算机的入站流量。
*	所有流量	All	全部	0.0.0.0/0	DENY	拒绝所有其他入站流量。

下面是与子网关联的网络 ACL 的示例出站规则。网络 ACL 没有任何状态。因此，您必须包含允许对入站流量做出响应的规则。

规则 #	类型	协议	端口范围	目的地	允许/拒绝	注释
100	自定义 TCP	TCP	1024-65535	*172.31.1.2/32*	允许	允许到远程计算机的出站响应。
*	所有流量	All	全部	0.0.0.0/0	拒绝	拒绝所有其他出站流量。

安全组规则

下表是与实例关联的安全组的示例入站规则。这些规则适用于与安全组关联的所有实例。拥有私钥（适用于与实例关联的密钥对）的用户可以使用 SSH 从远程计算机连接到实例。

协议类型	协议	端口范围	源	注释
所有流量	All	All	*sg-1234567890abcdef0*	允许在与此安全组关联的实例之间进行通信。
SSH	TCP	22	*172.31.1.2/32*	允许从远程计算机进行入站 SSH 访问。

下表是与实例关联的安全组的示例出站规则。安全组是有状态的。因此，您不需要允许对入站流量做出响应的规则。

协议类型	协议	端口范围	目的地	注释
所有流量	All	All	*sg-1234567890abcdef0*	允许在与此安全组关联的实例之间进行通信。

网络 ACL 和安全组之间的区别

下表总结了网络 ACL 和安全组之间的基本区别。

特征	网络 ACL	安全组
操作级别	子网级别	实例级别
范围	适用于关联子网中的所有实例	适用于与安全组关联的所有实例
Rule type	允许和拒绝规则	仅允许规则
规则评估	按升序评估规则，直到找到与流量匹配的规则	在决定是否允许流量前评估所有规则
返回流量	必须明确允许（无状态）	自动允许（有状态）

Amazon Virtual Private Cloud 中的恢复能力

AWS 全球基础设施围绕 AWS 区域和可用区构建。AWS 区域具有多个在物理上独立且隔离的可用区，这些可用区通过低延迟、高吞吐量、高度冗余的网络相互连接。利用可用区，您可以设计和操作在可用区之间无中断地自动实现失效转移的应用程序和数据库。与传统的单个或多个数据中心基础结构相比，可用区具有更高的可用性、容错性和可扩展性。

AWS 区域 是主要构建块，每个构建块都代表一个不同的地理位置，其中存放了多个在物理上独立且隔离的可用区。这些可用区通过低延迟、高吞吐量和高冗余的联网结构连接在一起，实现了在这些可用区之间的无缝通信和数据传输。

可用区的架构是一个关键的差异化因素，因为与传统的单个或多个数据中心基础设施相比，这些可用区的设计具有显著增强的稳定性和容错性。通过将资源分配到一个区域内的多个可用区，即可将应用程序和数据库设计为在区域之间自动进行失效转移，而不会对服务造成任何中断。这种级别的冗余和高可用性是任务关键型工作负载的关键要求，让组织能够构建弹性的云原生解决方案。

此外，AWS 基础设施的规模和全球覆盖范围使客户能够将应用程序部署在离最终用户更近的地方，从而减少延迟并改善用户整体体验。由于客户可以在其特定的监管及业务需求所要求的地理边界内存储和处理数据，全球多个区域的可用性还有助于实现有效的数据主权和合规性。

通过利用 AWS 全球基础设施，各组织可以构建自己的云环境，使其具有高可用性、容错性和可扩展性，并且可以灵活地适应不断变化的需求和不断发展的业务需求。这一坚实的基础是成功实现基于云的现代应用程序和服务的关键推动力。

有关 AWS 区域和可用区的更多信息，请参阅 AWS 全球基础结构。

您可以配置 VPC 以满足工作负载的故障恢复要求。有关更多信息，请参阅下列内容：

- 了解故障恢复模式和权衡（AWS 架构博客）
- 规划网络拓扑结构（AWS Well-Architected Framework）
- Amazon Virtual Private Cloud 连接选项（AWS 白皮书）

Amazon Virtual Private Cloud 的合规性验证

要了解某个 AWS 服务是否在特定合规性计划范围内，请参阅合规性计划范围内的 AWS 服务，然后选择您感兴趣的合规性计划。有关常规信息，请参阅 AWS 合规性计划、、。

您可以使用 AWS Artifact 下载第三方审计报告。有关更多信息，请参阅在 AWS Artifact 中下载报告、。

您使用 AWS 服务的合规性责任取决于您数据的敏感度、贵公司的合规性目标以及适用的法律法规。AWS 提供以下资源来帮助满足合规性：

- Security Compliance & Governance：这些解决方案实施指南讨论了架构考虑因素，并提供了部署安全性和合规性功能的步骤。
- 符合 HIPAA 要求的服务参考：列出符合 HIPAA 要求的服务。并非所有 AWS 服务 都符合 HIPAA 要求。
- AWS 合规性资源：此业务手册和指南集合可能适用于您的行业和位置。
- AWS 客户合规指南：从合规角度了解责任共担模式。这些指南总结了保护 AWS 服务 的最佳实践，并将指南映射到跨多个框架的安全控制，包括美国国家标准与技术研究院（NIST）、支付卡行业安全标准委员会（PCI）和国际标准化组织（ISO）。
- AWS Config 开发人员指南中的使用规则评估资源 - 此 AWS Config 服务评测您的资源配置对内部实践、行业指南和法规的遵循情况。
- AWS Security Hub：此 AWS 服务 向您提供 AWS 中安全状态的全面视图。Security Hub 通过安全控制措施评估您的 AWS 资源并检查其是否符合安全行业标准和最佳实践。有关受支持服务及控制措施的列表，请参阅 Security Hub 控制措施参考。

- Amazon GuardDuty：该 AWS 服务 通过监控您的环境中是否存在可疑和恶意活动，来检测您的 AWS 账户、工作负载、容器和数据面临的潜在威胁。GuardDuty 可以通过满足某些合规性框架规定的入侵检测要求，来协助您满足各种合规性要求，如 PCI DSS。
- AWS Audit Manager：此 AWS 服务 可帮助您持续审核您的 AWS 使用情况，以简化管理风险以及与相关法规和行业标准的合规性的方式。

屏蔽 VPC 和子网的公共访问权限

VPC 屏蔽公共访问权限（BPA）是一项集中式安全功能，可让您以权威方式屏蔽整个 AWS 账户对 VPC 资源的公共互联网访问，从而确保符合安全要求，同时为特定的例外情况和审计功能提供灵活性。

VPC BPA 功能有以下几种模式：

- 双向：进出此区域互联网网关和仅出口互联网网关的所有流量（排除的 VPC 和子网除外）均被阻止。
- 仅入口：此区域 VPC 的所有互联网流量（排除的 VPC 或子网除外）均被阻止。仅允许进出 NAT 网关和仅出口互联网网关的流量，因为这些网关仅允许建立出站连接。

您也可以针对不想阻止的流量为此功能创建“排除项”。排除是一种可以应用于单个 VPC 或子网的模式，可将其排除在账户的 VPC BPA 模式之外，并允许双向或仅出口访问。

排除可以采用以下任一模式：

- 双向：允许进出已排除 VPC 和子网的所有互联网流量。
- 仅出口：允许来自已排除 VPC 和子网的出站互联网流量。进入已排除 VPC 和子网的入站互联网流量已被阻止。这仅在 VPC BPA 设置为“双向”时适用。

内容

- VPC BPC BPA 基础知识
- 评测 VPC BPA 的影响并监控 VPC BPA
- 高级示例

VPC BPC BPA 基础知识

本节介绍有关 VPC BPA 的重要详细信息，包括哪些服务支持以及如何使用该功能。

内容

- 区域可用性
- AWS 服务影响和支持
- VPC BPA 限制
- 使用 IAM 策略控制对 VPC BPA 的访问
- 为您的账户启用 VPC BPA 双向模式
- 将 VPC BPA 模式更改为仅入口
- 创建和删除排除项
- 在组织级别启用 VPC BPA

区域可用性

VPC BPA 已在所有商业 AWS 区域提供，包括 GovCloud 和中国区域。

在本指南中，您还将找到有关将网络访问分析器和 Reachability Analyzer 与 VPC BPA 配合使用的信息。请注意，网络访问分析器和 Reachability Analyzer 并非在所有商业区域提供。有关网络访问分析器和 Reachability Analyzer 区域可用性的信息，请参阅《Network Access Analyzer Guide》中的 Limitations 和《Reachability Analyzer Guide》中的 Considerations。

AWS 服务影响和支持

以下资源和服务支持 VPC BPA，并且这些服务和资源的流量受 VPC BPA 的影响：

- 互联网网关：所有入站和出站流量都被阻止。
- 仅出口互联网网关：所有出站流量均被阻止。仅出口互联网网关不允许入站流量。
- 网关负载均衡器（GWLB）：即使排除了包含 GWLB 端点的子网，所有入站和出站流量也会被阻止。
- NAT 网关：所有入站和出站流量均被阻止。NAT 网关需要互联网网关才能连接互联网。
- 面向互联网的网络负载均衡器：所有入站和出站流量均被阻止。面向互联网的网络负载均衡器需要互联网网关才能连接互联网。
- 面向互联网的应用程序负载均衡器：所有入站和出站流量均被阻止。面向互联网的应用程序负载均衡器需要互联网网关才能连接互联网。
- Amazon CloudFront VPC 源：所有入站和出站流量均被阻止。

- AWS Direct Connect：所有使用公共虚拟接口（公有 IPv4 地址或全球单播 IPv6 地址）的入站和出站流量都将被阻止。此流量使用互联网网关（或仅出口互联网网关）进行连接。
- AWS 全球加速器：无论目标是否可以通过互联网访问，发往 VPC 的入站流量都会被阻止。
- AWS Network Firewall：即使排除了包含防火墙端点的子网，所有入站和出站流量也会被阻止。
- AWS Wavelength 运营商网关：所有入站和出站流量均被阻止。

VPC BPA 不会阻止或影响与私有连接相关的流量，如以下服务和资源的流量：

- AWS Client VPN
- AWS CloudWAN
- AWS Outposts 本地网关
- AWS Site-to-Site VPN
- Transit Gateway
- AWS Verified Access

> **Important**
>
> - 如果通过子网中 EC2 实例上运行的设备（例如第三方安全或监控工具）路由传入和传出流量，则在使用 VPC BPA 时，该子网需要排除流入和流出的流量。向设备子网而非互联网网关发送流量的其他子网不需要添加为排除项。
> - 即使启用了 VPC BPA，也允许流量从 VPC 中的资源私下发送到 VPC 中运行的其他服务（例如 EC2 DNS 解析程序或 Amazon OpenSearch Service），因为该流量不会经过 VPC 中的互联网网关。例如，这些服务可能会代表您向 VPC 之外的资源发出请求以解决 DNS 查询，如果不通过其他安全控制措施缓解，则可能会泄露有关您的 VPC 内资源活动的信息。

VPC BPA 限制

不允许 NAT 网关和仅出口互联网网关的本地区域（LZ）中不支持 VPC BPA 仅入口模式。

使用 IAM 策略控制对 VPC BPA 的访问

有关允许/拒绝访问 VPC BPA 功能的 IAM 策略的示例，请参阅 屏蔽 VPC 和子网的公共访问权限。

为您的账户启用 VPC BPA 双向模式

VPC BPA 双向模式阻止进出此区域互联网网关和仅出口互联网网关的所有流量（已排除 VPC 和子网除外）。有关排除项的更多信息，请参阅创建和删除排除项。

> **Important**
>
> 强烈建议您在生产账户中启用 VPC BPA 之前，仔细检查需要访问互联网的工作负载。

> **Note**
>
> - 要在您账户中的 VPC 和子网上启用 VPC BPA，您必须拥有 VPC 和子网。
> - 如果您当前与其他账户共享 VPC 子网，则子网所有者强制执行的 VPC BPA 模式也适用于参与者流量，但参与者无法控制影响共享子网的 VPC BPA 设置。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在左侧导航窗格中，选择设置。
3. 选择编辑公共访问权限设置。
4. 选择开启屏蔽公共访问和双向，然后选择保存更改。
5. 等待状态更改为开启。VPC BPA 设置生效和状态更新可能需要几分钟时间。

VPC BPA 双向模式现已开启。

AWS CLI

1. 启用 VPC BPA：

```
aws ec2 --region us-east-2 modify-vpc-block-public-access-options --internet-gateway-block-mode block-bidirectional
```

VPC BPA 设置生效和状态更新可能需要几分钟时间。

2. 查看 VPC BPA 的状态：

```
aws ec2 --region us-east-2 describe-vpc-block-public-access-options
```

将 VPC BPA 模式更改为仅入口

VPC BPA 仅入口模式会阻止此区域 VPC 的所有互联网流量（已排除的 VPC 或子网除外）。仅允许进出 NAT 网关和仅出口互联网网关的流量，因为这些网关仅允许建立出站连接。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在左侧导航窗格中，选择设置。
3. 选择编辑公共访问权限设置。
4. 将方向更改为仅入口。
5. 保存更改并等待状态更新。VPC BPA 设置生效和状态更新可能需要几分钟时间。

AWS CLI

1. 修改 VPC BPA 阻止方向：

   ```
   aws ec2 --region us-east-2 modify-vpc-block-public-access-options --internet-gateway-block-mode block-ingress
   ```

   VPC BPA 设置生效和状态更新可能需要几分钟时间。
2. 查看 VPC BPA 的状态：

   ```
   aws ec2 --region us-east-2 describe-vpc-block-public-access-options
   ```

创建和删除排除项

VPC BPA 排除是一种可以应用于单个 VPC 或子网的模式，可将其排除在账户的 VPC BPA 模式之外，并允许双向或仅出口访问。即使账户未启用 VPC BPA，您也可以为 VPC 和子网创建 VPC BPA 排除项，以确保启用 VPC BPA 时排除项不会中断流量。VPC 的排除会自动应用于 VPC 中的所有子网。

您最多可以创建 50 个排除项。有关请求提高限制的信息，请参阅 Amazon VPC 配额中每个账户的 VPC BPA 排除项。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在左侧导航窗格中，选择设置。
3. 在阻止公开访问选项卡的排除下，执行以下任一操作：
   - 要删除排除项，请选择排除项，然后选择操作 > 删除排除项。
   - 要创建排除项，请选择创建排除项并继续执行后续步骤。
4. 选择阻止方向：
   - 双向：允许进出已排除 VPC 和子网的所有互联网流量。
   - 仅出口：允许来自已排除 VPC 和子网的出站互联网流量。阻止进入已排除 VPC 和子网的入站互联网流量。当 VPC BPA 设置为双向时，此设置适用。
5. 选择 VPC 或子网。
6. 选择创建排除项。
7. 等待排除项状态变为活动。您可能需要刷新排除项表才能查看更改。

排除项已创建。

AWS CLI

1. 修改排除允许方向：

```
aws ec2 --region us-east-2 create-vpc-block-public-access-exclusion --subnet-
id subnet-id --internet-gateway-exclusion-mode allow-bidirectional
```

2. 更新排除项状态可能需要一段时间。要查看排除项的状态：

```
aws ec2 --region us-east-2 describe-vpc-block-public-access-exclusions --
exclusion-ids exclusion-id
```

在组织级别启用 VPC BPA

如果您使用 AWS Organizations 来管理组织中的账户，则可以使用 AWS Organizations 声明式策略对组织中的账户强制执行 VPC BPA。有关 VPC BPA 声明式策略的更多信息，请参阅《AWS Organizations 用户指南》中的 Supported declarative policies。

> **Note**
>
> - 您可以使用 VPC BPA 声明式策略配置是否允许排除，但不能使用该策略创建排除项。若要创建排除项，仍然需要在拥有 VPC 的账户中创建排除项。有关创建 VPC BPA 排除项的更多信息，请参阅创建和删除排除项。
> - 如果启用了 VPC BPA 声明式策略，则在阻止公开访问设置中，您将看到由声明式策略管理，并且您将无法在账户级别修改 VPC BPA 设置。

评测 VPC BPA 的影响并监控 VPC BPA

本节包含有关如何在启用 VPC BPA 之前评测其影响及如何在启用 VPC BPA 之后监控流量是否被阻止的信息。

内容

- 使用网络访问分析器评测 VPC BPA 的影响
- 使用流日志监控 VPC BPA 影响
- 使用 CloudTrail 追踪排除项删除
- 使用 Reachability Analyzer 验证连接是否被阻止

使用网络访问分析器评测 VPC BPA 的影响

在本节中，在启用 VPC BPA 并阻止访问之前，您将使用网络访问分析器查看账户中使用互联网网关的资源。使用此分析可了解在您的账户中启用 VPC BPA 和阻止流量所产生的影响。

> **Note**
>
> - 网络访问分析器不支持 IPv6；因此您将无法使用它来查看 VPC BPA 对仅出口互联网网关出站 IPv6 流量的潜在影响。

- 您需要为使用网络访问分析器执行的分析付费。有关更多信息，请参阅《网络访问分析器 Guide》中的 Pricing。
- 有关网络访问分析器区域可用性的信息，请参阅《Network Access Analyzer Guide》中的 Limitations。

AWS Management Console

1. 打开位于 https://console.aws.amazon.com/networkinsights/ 的 AWS 网络见解控制台。
2. 选择网络访问分析器。
3. 选择创建网络访问范围。
4. 选择评估 VPC 阻止公开访问的影响，然后选择下一步。
5. 该模板已配置为分析您账户中进出互联网网关的流量。您可以在来源和目标下查看此项。
6. 选择下一步。
7. 选择创建网络访问范围。
8. 选择您刚创建的范围，然后选择分析。
9. 等待分析完成。
10. 查看分析的调查发现。调查发现下的每一行都显示数据包在网络中进出您账户中的互联网网关可以采用的网络路径。在这种情况下，如果您启用 VPC BPA，并且这些调查发现中显示的 VPC 和/或子网均未配置为 VPC BPA 排除项，则流向这些 VPC 和子网的流量将受到限制。
11. 分析每项调查发现，了解 VPC BPA 对 VPC 中资源的影响。

影响分析已完成。

AWS CLI

1. 创建网络访问范围：

```
aws ec2 create-network-insights-access-scope --region us-east-2 --match-paths
 "Source={ResourceStatement={ResourceTypes=["AWS::EC2::InternetGateway"]}}"
 "Destination={ResourceStatement={ResourceTypes=["AWS::EC2::InternetGateway"]}}"
```

2. 开始范围分析：

```
aws ec2 start-network-insights-access-scope-analysis  --region us-east-2 --
network-insights-access-scope-id nis-id
```

3. 获取分析的结果：

```
aws ec2 get-network-insights-access-scope-analysis-findings  --region us-east-2
 --network-insights-access-scope-analysis-id nisa-0aa383a1938f94cd1 --max-items
 1
```

结果显示进出您账户所有 VPC 中互联网网关的流量。结果被归类为“调查发现”。"FindingId": "AnalysisFinding-1" 表示这是分析中的第一个调查发现。请注意，有多个调查发现，每个调查发现都表明启用 VPC BPA 将影响流量。第一个调查发现表明流量从互联网网关（"SequenceNumber": 1）开始，依次传递到 NACL（"SequenceNumber": 2）、安全组（"SequenceNumber": 3），然后在实例（"SequenceNumber": 4）处结束。

4. 分析调查发现以了解 VPC BPA 对 VPC 中资源的影响。

影响分析已完成。

使用流日志监控 VPC BPA 影响

利用 VPC 流日志这项功能，您可以捕获有关传入和传出您的 VPC 中弹性网络接口的 IP 流量的信息。您可以使用此功能监控被 VPC BPA 阻止的流量到达您的实例网络接口。

使用使用流日志中的步骤为您的 VPC 创建流日志。

创建流日志时，请确保使用包含字段 `reject-reason` 的自定义格式。

查看流日志时，如果发往 ENI 的流量因 VPC BPA 而被拒绝，您将在流日志条目中看到 BPA 的 `reject-reason`。

除了 VPC 流日志的标准限制外，请注意以下 VPC BPA 特定的限制：

- VPC BPA 的流日志不包括跳过的记录。
- 即使您在流日志中包含 `bytes` 字段，VPC BPA 的流日志也不会包含 `bytes`。

使用 CloudTrail 追踪排除项删除

本节向您介绍如何使用 AWS CloudTrail 监控和追踪 VPC BPA 排除项的删除。

AWS Management Console

通过在位于 https://console.aws.amazon.com/cloudtrailv2/ 的 AWS CloudTrail 控制台中查找资源类型 > `AWS::EC2::VPCBlockPublicAccessExclusion`，您可以在 CloudTrail 事件历史记录中查看任何已删除的排除项。

AWS CLI

您可以使用 `lookup-events` 命令查看与删除排除项有关的事件：

```
aws cloudtrail lookup-events --lookup-attributes
 AttributeKey=ResourceType,AttributeValue=AWS::EC2::VPCBlockPublicAccessExclusion
```

使用 Reachability Analyzer 验证连接是否被阻止

VPC Reachability Analyzer 可用于评估根据您的网络配置（包括 VPC BPA 设置）是否可以访问某些网络路径。

有关 Reachability Analyzer 区域可用性的信息，请参阅《Reachability Analyzer Guide》中的 Considerations。

AWS Management Console

1. 打开位于 https://console.aws.amazon.com/networkinsights/home#ReachabilityAnalyzer 的 AWS 网络见解控制台。
2. 单击创建和分析路径。
3. 对于源类型，选择互联网网关，然后从源下拉列表中选择要阻止流量的互联网网关。
4. 对于目标类型，选择实例，然后从目标下拉列表中选择要阻止流量的实例。
5. 单击创建和分析路径。
6. 等待分析完成。这可能需要几分钟时间。
7. 完成后，您应该会看到可访问性状态为无法访问，并且路径详细信息显示 `VPC_BLOCK_PUBLIC_ACCESS_ENABLED` 是导致此可访问性问题的原因。

AWS CLI

1. 使用要阻止流量传入的互联网网关 ID（源）和要阻止流量传出的实例 ID（目标）创建网络路径：

```
aws ec2 --region us-east-2 create-network-insights-path --source igw-id --
destination instance-id --protocol TCP
```

2. 开始对网络路径进行分析：

```
aws ec2 --region us-east-2 start-network-insights-analysis --network-insights-
path-id nip-id
```

3. 检索分析的结果：

```
aws ec2 --region us-east-2 describe-network-insights-analyses --network-
insights-analysis-ids nia-id
```

4. 请确认 VPC_BLOCK_PUBLIC_ACCESS_ENABLED 是无法访问的 ExplanationCode。

高级示例

本节包含一个高级示例，可帮助您了解 VPC 屏蔽公共访问权限功能在不同方案中如何工作。每个方案都以之前的方案为基础构建，因此按顺序完成这些步骤很重要。

Important

请勿在生产账户中完成此示例。强烈建议您在生产账户中启用 VPC BPA 之前，仔细检查需要访问互联网的工作负载。

Note

要充分了解 VPC BPA 功能，您的账户中需要某些资源。在本节中，我们提供一个 AWS CloudFormation 模板，您可以使用该模板来预置所需的资源，以充分了解此功能如何工作。使用 CloudFormation 模板预置的资源以及使用网络访问分析器和 Reachability Analyzer 执行分析会产生相关的费用。如果您使用本节中的模板，则请确保在完成此示例后完成清理步骤。

内容

- 部署 CloudFormation 模板（可选）
- 使用网络访问分析器查看 VPC BPA 的影响

- 场景 1 - 连接到未启用 VPC BPA 的实例
- 场景 2 - 启用 VPC BPA 双向模式
- 场景 3 - 将 VPC BPA 更改为仅入口模式
- 场景 4 - 创建排除项
- 场景 5 - 修改排除模式
- 场景 6 - 修改 VPC BPA 模式
- 清理

部署 CloudFormation 模板（可选）

要演示此功能如何工作，您需要一个 VPC、子网、实例和其他资源。为了更轻松地完成本演示，我们在下面提供了一个 AWS CloudFormation 模板，您可以使用该模板来快速启动本演示中方案所需的资源。这一步骤是可选的，您可能只是想查看本节场景中的图表。

> **Note**
>
> - 在本节中使用 CloudFormation 模板创建的资源会产生一些相关费用，例如 NAT 网关和公有 IPv4 地址的费用。为避免产生额外的费用，请确保完成清理步骤，移除为本示例创建的所有资源。
> - 此 CloudFormation 模板创建了 VPC BPA 所需的底层资源，但其本身未启用 VPC BPA 功能。在选择单独启用 VPC BPA 功能后，此处部署的资源旨在帮助您了解和测试这项功能。

该模板在您的账户中创建以下资源：

- 仅出口互联网网关
- 互联网网关
- NAT 网关
- 两个公有子网
- 一个私有子网
- 具有公有和私有 IPv4 地址的两个实例
- 具有 IPv6 地址和私有 IPv4 地址的一个 EC2 实例
- 仅具有私有 IPv4 地址的一个 EC2 实例

- 允许 SSH 和 ICMP 入站流量且允许所有出站流量的安全组
- VPC 流日志
- 子网 B 中的一个 EC2 Instance Connect 端点

复制下面的模板并将其保存到 .yaml 文件。

```
AWSTemplateFormatVersion: '2010-09-09'
Description: Creates a VPC with public and private subnets, NAT gateway, and EC2
 instances for VPC BPA.

Parameters:
  InstanceAMI:
    Description: ID of the Amazone Machine Image (AMI) to use with the instances
 launched by this template
    Type: AWS::EC2::Image::Id
  InstanceType:
    Description: EC2 Instance type to use with the instances launched by this template
    Type: String
    Default: t2.micro

Resources:

  # VPC
  VPCBPA:
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: 10.0.0.0/16
      EnableDnsHostnames: true
      EnableDnsSupport: true
      InstanceTenancy: default
      Tags:
        - Key: Name
          Value: VPC BPA

  # VPC IPv6 CIDR
  VPCBPAIpv6CidrBlock:
    Type: AWS::EC2::VPCCidrBlock
    Properties:
      VpcId: !Ref VPCBPA
      AmazonProvidedIpv6CidrBlock: true

  # EC2 Key Pair
```

```
  VPCBPAKeyPair:
    Type: AWS::EC2::KeyPair
    Properties:
      KeyName: vpc-bpa-key

  # Internet Gateway
  VPCBPAInternetGateway:
    Type: AWS::EC2::InternetGateway
    Properties:
      Tags:
        - Key: Name
          Value: VPC BPA Internet Gateway

  VPCBPAInternetGatewayAttachment:
    Type: AWS::EC2::VPCGatewayAttachment
    Properties:
      VpcId: !Ref VPCBPA
      InternetGatewayId: !Ref VPCBPAInternetGateway

  # Egress-Only Internet Gateway
  VPCBPAEgressOnlyInternetGateway:
    Type: AWS::EC2::EgressOnlyInternetGateway
    Properties:
      VpcId: !Ref VPCBPA

  # Subnets
  VPCBPAPublicSubnetA:
    Type: AWS::EC2::Subnet
    Properties:
      VpcId: !Ref VPCBPA
      CidrBlock: 10.0.1.0/24
      MapPublicIpOnLaunch: true
      Tags:
        - Key: Name
          Value: VPC BPA Public Subnet A

  VPCBPAPublicSubnetB:
    Type: AWS::EC2::Subnet
    Properties:
      VpcId: !Ref VPCBPA
      CidrBlock: 10.0.2.0/24
      MapPublicIpOnLaunch: true
      Tags:
        - Key: Name
```

```
        Value: VPC BPA Public Subnet B

  VPCBPAPrivateSubnetC:
    Type: AWS::EC2::Subnet
    Properties:
      VpcId: !Ref VPCBPA
      CidrBlock: 10.0.3.0/24
      MapPublicIpOnLaunch: false
      Ipv6CidrBlock: !Select [0, !GetAtt VPCBPA.Ipv6CidrBlocks]
      AssignIpv6AddressOnCreation: true
      Tags:
        - Key: Name
          Value: VPC BPA Private Subnet C

  # NAT Gateway
  VPCBPANATGateway:
    Type: AWS::EC2::NatGateway
    Properties:
      AllocationId: !GetAtt VPCBPANATGatewayEIP.AllocationId
      SubnetId: !Ref VPCBPAPublicSubnetB
      Tags:
        - Key: Name
          Value: VPC BPA NAT Gateway

  VPCBPANATGatewayEIP:
    Type: AWS::EC2::EIP
    Properties:
      Domain: vpc
      Tags:
        - Key: Name
          Value: VPC BPA NAT Gateway EIP

  # Route Tables
  VPCBPAPublicRouteTable:
    Type: AWS::EC2::RouteTable
    Properties:
      VpcId: !Ref VPCBPA
      Tags:
        - Key: Name
          Value: VPC BPA Public Route Table

  VPCBPAPublicRoute:
    Type: AWS::EC2::Route
    DependsOn: VPCBPAInternetGatewayAttachment
```

```
    Properties:
      RouteTableId: !Ref VPCBPAPublicRouteTable
      DestinationCidrBlock: 0.0.0.0/0
      GatewayId: !Ref VPCBPAInternetGateway

  VPCBPAPublicSubnetARouteTableAssoc:
    Type: AWS::EC2::SubnetRouteTableAssociation
    Properties:
      SubnetId: !Ref VPCBPAPublicSubnetA
      RouteTableId: !Ref VPCBPAPublicRouteTable

  VPCBPAPublicSubnetBRouteTableAssoc:
    Type: AWS::EC2::SubnetRouteTableAssociation
    Properties:
      SubnetId: !Ref VPCBPAPublicSubnetB
      RouteTableId: !Ref VPCBPAPublicRouteTable

  VPCBPAPrivateRouteTable:
    Type: AWS::EC2::RouteTable
    Properties:
      VpcId: !Ref VPCBPA
      Tags:
        - Key: Name
          Value: VPC BPA Private Route Table

  VPCBPAPrivateRoute:
    Type: AWS::EC2::Route
    Properties:
      RouteTableId: !Ref VPCBPAPrivateRouteTable
      DestinationCidrBlock: 0.0.0.0/0
      NatGatewayId: !Ref VPCBPANATGateway

  VPCBPAPrivateSubnetCRoute:
    Type: AWS::EC2::Route
    Properties:
      RouteTableId: !Ref VPCBPAPrivateRouteTable
      DestinationIpv6CidrBlock: ::/0
      EgressOnlyInternetGatewayId: !Ref VPCBPAEgressOnlyInternetGateway

  VPCBPAPrivateSubnetCRouteTableAssociation:
    Type: AWS::EC2::SubnetRouteTableAssociation
    Properties:
      SubnetId: !Ref VPCBPAPrivateSubnetC
      RouteTableId: !Ref VPCBPAPrivateRouteTable
```

```
# EC2 Instances Security Group
VPCBPAInstancesSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupName: VPC BPA Instances Security Group
    GroupDescription: Allow SSH and ICMP access
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        CidrIp: 0.0.0.0/0
      - IpProtocol: icmp
        FromPort: -1
        ToPort: -1
        CidrIp: 0.0.0.0/0
    VpcId: !Ref VPCBPA
    Tags:
      - Key: Name
        Value: VPC BPA Instances Security Group

# EC2 Instances
VPCBPAInstanceA:
  Type: AWS::EC2::Instance
  Properties:
    ImageId: !Ref InstanceAMI
    InstanceType: t2.micro
    KeyName: !Ref VPCBPAKeyPair
    SubnetId: !Ref VPCBPAPublicSubnetA
    SecurityGroupIds:
      - !Ref VPCBPAInstancesSecurityGroup
    Tags:
      - Key: Name
        Value: VPC BPA Instance A

VPCBPAInstanceB:
  Type: AWS::EC2::Instance
  Properties:
    ImageId: !Ref InstanceAMI
    InstanceType: !Ref InstanceType
    KeyName: !Ref VPCBPAKeyPair
    SubnetId: !Ref VPCBPAPublicSubnetB
    SecurityGroupIds:
      - !Ref VPCBPAInstancesSecurityGroup
```

```
      Tags:
        - Key: Name
          Value: VPC BPA Instance B

  VPCBPAInstanceC:
    Type: AWS::EC2::Instance
    Properties:
      ImageId: !Ref InstanceAMI
      InstanceType: !Ref InstanceType
      KeyName: !Ref VPCBPAKeyPair
      SubnetId: !Ref VPCBPAPrivateSubnetC
      SecurityGroupIds:
        - !Ref VPCBPAInstancesSecurityGroup
      Tags:
        - Key: Name
          Value: VPC BPA Instance C

  VPCBPAInstanceD:
    Type: AWS::EC2::Instance
    Properties:
      ImageId: !Ref InstanceAMI
      InstanceType: !Ref InstanceType
      KeyName: !Ref VPCBPAKeyPair
      NetworkInterfaces:
        - DeviceIndex: '0'
          GroupSet:
            - !Ref VPCBPAInstancesSecurityGroup
          SubnetId: !Ref VPCBPAPrivateSubnetC
          Ipv6AddressCount: 1
      Tags:
        - Key: Name
          Value: VPC BPA Instance D

  # Flow Logs IAM Role
  VPCBPAFlowLogRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Version: '2012-10-17'
        Statement:
          - Effect: Allow
            Principal:
              Service: vpc-flow-logs.amazonaws.com
            Action: 'sts:AssumeRole'
```

```
      Tags:
        - Key: Name
          Value: VPC BPA Flow Logs Role

  VPCBPAFlowLogPolicy:
    Type: AWS::IAM::Policy
    Properties:
      PolicyName: VPC-BPA-FlowLogsPolicy
      PolicyDocument:
        Version: '2012-10-17'
        Statement:
          - Effect: Allow
            Action:
              - 'logs:CreateLogGroup'
              - 'logs:CreateLogStream'
              - 'logs:PutLogEvents'
              - 'logs:DescribeLogGroups'
              - 'logs:DescribeLogStreams'
            Resource: '*'
      Roles:
        - !Ref VPCBPAFlowLogRole

  # Flow Logs
  VPCBPAFlowLog:
    Type: AWS::EC2::FlowLog
    Properties:
      ResourceId: !Ref VPCBPA
      ResourceType: VPC
      TrafficType: ALL
      LogDestinationType: cloud-watch-logs
      LogGroupName: /aws/vpc-flow-logs/VPC-BPA
      DeliverLogsPermissionArn: !GetAtt VPCBPAFlowLogRole.Arn
      LogFormat: '${version} ${account-id} ${interface-id} ${srcaddr} ${dstaddr}
 ${srcport} ${dstport} ${protocol} ${packets} ${bytes} ${start} ${end} ${action} ${log-
status} ${vpc-id} ${subnet-id} ${instance-id} ${tcp-flags} ${type} ${pkt-srcaddr}
 ${pkt-dstaddr} ${region} ${az-id} ${sublocation-type} ${sublocation-id} ${pkt-src-aws-
service} ${pkt-dst-aws-service} ${flow-direction} ${traffic-path} ${reject-reason}'
      Tags:
        - Key: Name
          Value: VPC BPA Flow Logs

  # EC2 Instance Connect Endpoint
  VPCBPAEC2InstanceConnectEndpoint:
    Type: AWS::EC2::InstanceConnectEndpoint
```

```
    Properties:
      SecurityGroupIds:
        - !Ref VPCBPAInstancesSecurityGroup
      SubnetId: !Ref VPCBPAPublicSubnetB

Outputs:
  VPCBPAVPCId:
    Description: A reference to the created VPC
    Value: !Ref VPCBPA
    Export:
      Name: vpc-id

  VPCBPAPublicSubnetAId:
    Description: The ID of the public subnet A
    Value: !Ref VPCBPAPublicSubnetA

  VPCBPAPublicSubnetAName:
    Description: The name of the public subnet A
    Value: VPC BPA Public Subnet A

  VPCBPAPublicSubnetBId:
    Description: The ID of the public subnet B
    Value: !Ref VPCBPAPublicSubnetB

  VPCBPAPublicSubnetBName:
    Description: The name of the public subnet B
    Value: VPC BPA Public Subnet B

  VPCBPAPrivateSubnetCId:
    Description: The ID of the private subnet C
    Value: !Ref VPCBPAPrivateSubnetC

  VPCBPAPrivateSubnetCName:
    Description: The name of the private subnet C
    Value: VPC BPA Private Subnet C

  VPCBPAInstanceAId:
    Description: The ID of instance A
    Value: !Ref VPCBPAInstanceA

  VPCBPAInstanceBId:
    Description: The ID of instance B
    Value: !Ref VPCBPAInstanceB
```

```
VPCBPAInstanceCId:
  Description: The ID of instance C
  Value: !Ref VPCBPAInstanceC

VPCBPAInstanceDId:
  Description: The ID of instance D
  Value: !Ref VPCBPAInstanceD
```

AWS Management Console

1. 从 AWS CloudFormation 打开 https://console.aws.amazon.com/cloudformation/ 控制台。
2. 选择创建堆栈并上传 .yaml 模板文件。
3. 完成启动模板的步骤。您需要输入映像 ID 和实例类型（如 t2.micro）。您还需要允许 CloudFormation 为您创建一个 IAM 角色，以创建流日志并获得登录 Amazon CloudWatch 权限。
4. 启动堆栈后，请查看事件选项卡以查看进度，并确保堆栈已完成，然后再继续。

AWS CLI

1. 运行以下命令创建 CloudFormation 堆栈：

   ```
   aws cloudformation create-stack --stack-name VPC-BPA-stack --template-body
    file://sampletemplate.yaml --capabilities CAPABILITY_IAM --region us-east-2
   ```

   输出：

   ```
   {
       "StackId": "arn:aws:cloudformation:us-east-2:470889052923:stack/VPC-BPA-
   stack/8a7a2cc0-8001-11ef-b196-06386a84b72f"
   }
   ```

2. 查看进度并确保堆栈已完成，然后再继续：

   ```
   aws cloudformation describe-stack-events --stack-name VPC-BPA-stack --region us-
   east-2
   ```

使用网络访问分析器查看 VPC BPA 的影响

在本节中，您将使用网络访问分析器查看账户中使用互联网网关的资源。使用此分析可了解在您的账户中启用 VPC BPA 和阻止流量所产生的影响。

有关网络访问分析器区域可用性的信息，请参阅《Network Access Analyzer Guide》中的 Limitations。

AWS Management Console

1. 打开位于 https://console.aws.amazon.com/networkinsights/ 的 AWS 网络见解控制台。
2. 选择网络访问分析器。
3. 选择创建网络访问范围。
4. 选择评估 VPC 阻止公开访问的影响，然后选择下一步。
5. 该模板已配置为分析您账户中进出互联网网关的流量。您可以在来源和目标下查看此项。
6. 选择下一步。
7. 选择创建网络访问范围。
8. 选择您刚创建的范围，然后选择分析。
9. 等待分析完成。
10. 查看分析的调查发现。调查发现下的每一行都显示数据包在网络中进出您账户中的互联网网关可以采用的网络路径。在这种情况下，如果您启用 VPC BPA，并且这些调查发现中显示的 VPC 和/或子网均未配置为 VPC BPA 排除项，则流向这些 VPC 和子网的流量将受到限制。
11. 分析每项调查发现，了解 VPC BPA 对 VPC 中资源的影响。

影响分析已完成。

AWS CLI

1. 创建网络访问范围：

```
aws ec2 create-network-insights-access-scope --match-paths
 "Source={ResourceStatement={ResourceTypes=["AWS::EC2::InternetGateway"]}}"
 "Destination={ResourceStatement={ResourceTypes=["AWS::EC2::InternetGateway"]}}"
 --region us-east-2
```

输出：

```
{
  "NetworkInsightsAccessScope": {
    "NetworkInsightsAccessScopeId": "nis-04cad3c4b3a1d5e3e",
    "NetworkInsightsAccessScopeArn": "arn:aws:ec2:us-
east-2:470889052923:network-insights-access-scope/nis-04cad3c4b3a1d5e3e",
    "CreatedDate": "2024-09-30T15:55:53.171000+00:00",
    "UpdatedDate": "2024-09-30T15:55:53.171000+00:00"
  },
  "NetworkInsightsAccessScopeContent": {
    "NetworkInsightsAccessScopeId": "nis-04cad3c4b3a1d5e3e",
    "MatchPaths": [
      {
        "Source": {
          "ResourceStatement": {
            "ResourceTypes": [
              "AWS::EC2::InternetGateway"
            ]
          }
        }
      },
      {
        "Destination": {
          "ResourceStatement": {
            "ResourceTypes": [
              "AWS::EC2::InternetGateway"
            ]
          }
        }
      }
    ]
  }
}
```

2. 开始范围分析：

```
aws ec2 start-network-insights-access-scope-analysis --network-insights-access-
scope-id nis-04cad3c4b3a1d5e3e --region us-east-2
```

输出：

```
{
  "NetworkInsightsAccessScopeAnalysis": {
```

```
    "NetworkInsightsAccessScopeAnalysisId": "nisa-0aa383a1938f94cd1",
    "NetworkInsightsAccessScopeAnalysisArn": "arn:aws:ec2:us-
east-2:470889052923:network-insights-access-scope-analysis/
nisa-0aa383a1938f94cd",
    "NetworkInsightsAccessScopeId": "nis-04cad3c4b3a1d5e3e",
    "Status": "running",
    "StartDate": "2024-09-30T15:56:59.109000+00:00",
    "AnalyzedEniCount": 0
  }
}
```

3. 获取分析的结果：

```
aws ec2 get-network-insights-access-scope-analysis-findings --network-insights-
access-scope-analysis-id nisa-0aa383a1938f94cd1 --region us-east-2 --max-items 1
```

输出：

```
{
  "AnalysisFindings": [
    {
      "NetworkInsightsAccessScopeAnalysisId": "nisa-0aa383a1938f94cd1",
      "NetworkInsightsAccessScopeId": "nis-04cad3c4b3a1d5e3e",
      "FindingId": "AnalysisFinding-1",
      "FindingComponents": [
        {
          "SequenceNumber": 1,
          "Component": {
            "Id": "igw-04a5344b4e30486f1",
            "Arn": "arn:aws:ec2:us-east-2:470889052923:internet-gateway/
igw-04a5344b4e30486f1",
            "Name": "VPC BPA Internet Gateway"
          },
          "OutboundHeader": {
            "DestinationAddresses": [
              "10.0.1.85/32"
            ]
          },
          "InboundHeader": {
            "DestinationAddresses": [
              "10.0.1.85/32"
            ],
            "DestinationPortRanges": [
```

```
            {
              "From": 22,
              "To": 22
            }
          ],
          "Protocol": "6",
          "SourceAddresses": [
            "0.0.0.0/5",
            "100.0.0.0/10",
            "96.0.0.0/6"
          ],
          "SourcePortRanges": [
            {
              "From": 0,
              "To": 65535
            }
          ]
        },
        "Vpc": {
          "Id": "vpc-0762547ec48b6888d",
          "Arn": "arn:aws:ec2:us-east-2:470889052923:vpc/
vpc-0762547ec48b6888d",
          "Name": "VPC BPA"
        }
      },
      {
        "SequenceNumber": 2,
        "AclRule": {
          "Cidr": "0.0.0.0/0",
          "Egress": false,
          "Protocol": "all",
          "RuleAction": "allow",
          "RuleNumber": 100
        },
        "Component": {
          "Id": "acl-06194fc3a4a03040b",
          "Arn": "arn:aws:ec2:us-east-2:470889052923:network-acl/
acl-06194fc3a4a03040b"
        }
      },
      {
        "SequenceNumber": 3,
        "Component": {
          "Id": "sg-093dde06415d03924",
```

```
            "Arn": "arn:aws:ec2:us-east-2:470889052923:security-group/
sg-093dde06415d03924",
            "Name": "VPC BPA Instances Security Group"
          },
          "SecurityGroupRule": {
            "Cidr": "0.0.0.0/0",
            "Direction": "ingress",
            "PortRange": {
              "From": 22,
              "To": 22
            },
            "Protocol": "tcp"
          }
        },
        {
          "SequenceNumber": 4,
          "AttachedTo": {
            "Id": "i-058db34f9a0997895",
            "Arn": "arn:aws:ec2:us-east-2:470889052923:instance/
i-058db34f9a0997895",
            "Name": "VPC BPA Instance A"
          },
          "Component": {
            "Id": "eni-0fa23f2766f03b286",
            "Arn": "arn:aws:ec2:us-east-2:470889052923:network-interface/
eni-0fa23f2766f03b286"
          },
          "InboundHeader": {
            "DestinationAddresses": [
              "10.0.1.85/32"
            ],
            "DestinationPortRanges": [
              {
                "From": 22,
                "To": 22
              }
            ],
            "Protocol": "6",
            "SourceAddresses": [
              "0.0.0.0/5",
              "100.0.0.0/10",
              "96.0.0.0/6"
            ],
            "SourcePortRanges": [
```

```
              {
                "From": 0,
                "To": 65535
              }
            ]
          },
          "Subnet": {
            "Id": "subnet-035d235a762eeed04",
            "Arn": "arn:aws:ec2:us-east-2:470889052923:subnet/
subnet-035d235a762eeed04",
            "Name": "VPC BPA Public Subnet A"
          },
          "Vpc": {
            "Id": "vpc-0762547ec48b6888d",
            "Arn": "arn:aws:ec2:us-east-2:470889052923:vpc/
vpc-0762547ec48b6888d",
            "Name": "VPC BPA"
          }
        }
      ]
    }
  ],
  "AnalysisStatus": "succeeded",
  "NetworkInsightsAccessScopeAnalysisId": "nisa-0aa383a1938f94cd1",
  "NextToken":
 "eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxfQ=="
}
```


结果显示进出您账户所有 VPC 中互联网网关的流量。结果被归类为“调查发现”。"FindingId": "AnalysisFinding-1" 表示这是分析中的第一个调查发现。请注意，有多个调查发现，每个调查发现都表明启用 VPC BPA 将影响流量。第一个调查发现表明流量从互联网网关（"SequenceNumber": 1）开始，依次传递到 NACL（"SequenceNumber": 2）、安全组（"SequenceNumber": 3），然后在实例（"SequenceNumber": 4）处结束。

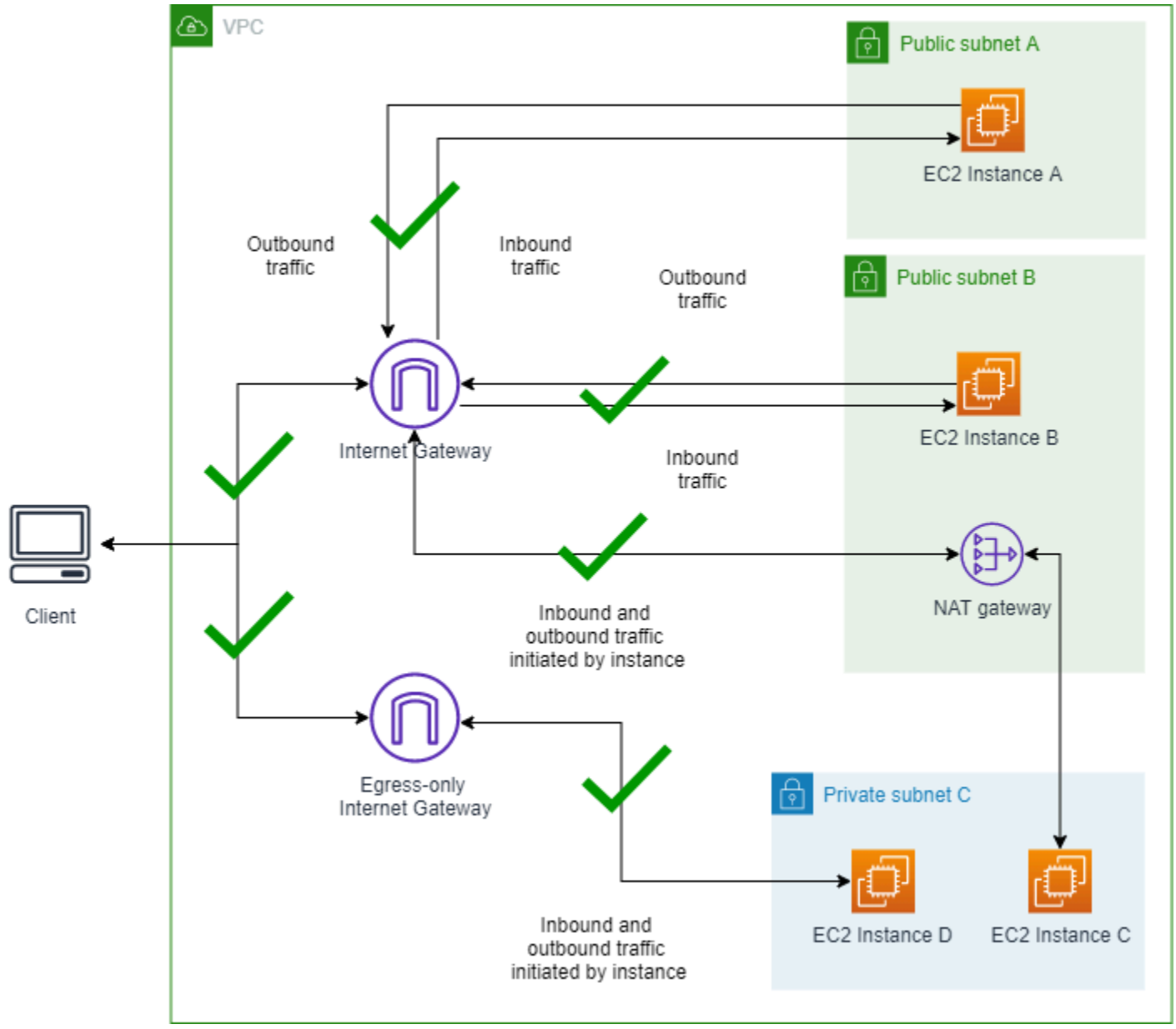
4. 分析调查发现以了解 VPC BPA 对 VPC 中资源的影响。

影响分析已完成。

场景 1 - 连接到未启用 VPC BPA 的实例

在本节中，可以通过允许入站和出站流量的互联网网关，从互联网访问公有子网 A 和 B 中的 EC2 实例。私有子网中的实例 C 和实例 D 可以通过 NAT 网关或仅出口互联网网关发出出站流量，但无法直接从互联网访问。此设置为某些资源提供 Internet 访问，同时保护其他资源。本设置旨在设置基准并确保在启用 VPC BPA 之前可以访问所有实例，您需要连接到所有实例并对公有 IP 地址运行 ping 命令。

未启用 VPC BPA 的 VPC 示意图：



1.1 连接到实例

完成本节以在关闭 VPC BPA 的情况下连接到您的实例，以确保可以毫无问题地进行连接。在本示例中使用 CloudFormation 创建的所有实例都有“VPC BPA 实例 A”之类的名称。

AWS Management Console

1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 打开实例 A 的详细信息。
3. 通过 EC2 Instance Connect > 使用 EC2 Instance Connect 端点连接选项连接到实例 A。
4. 选择连接。成功连接到该实例后，运行 ping www.amazon.com 命令确认您可以向互联网发送出站请求。
5. 使用与连接实例 A 相同的方法连接到实例 B、C 和 D。从每个实例 ping www.amazon.com 以验证您是否可以将出站请求发送到 Internet。

AWS CLI

1. 使用公有 IPv4 地址对实例 A 运行 Ping 命令以检查入站流量：

   ```
   ping 18.225.8.244
   ```

   输出：

   ```
   Pinging 18.225.8.244 with 32 bytes of data:

   Reply from 18.225.8.244: bytes=32 time=51ms TTL=110
   Reply from 18.225.8.244: bytes=32 time=61ms TTL=110
   ```

   请注意，运行 ping 命令成功并且流量未被阻止。

2. 使用私有 IPv4 地址连接并检查出站流量：

   ```
   aws ec2-instance-connect ssh --instance-id i-058db34f9a0997895 --region us-
   east-2 --connection-type eice
   ```

   输出：

   ```
   A newer release of "Amazon Linux" is available. Version 2023.5.20240916:
   Run "/usr/bin/dnf check-release-update" for full release and version update info
   ```

```
,     #_   ~_  ####_        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~._.   _/
/ /
/m/'
Last login: Fri Sep 27 18:27:57 2024 from 3.16.146.5
[ec2-user@ip-10-0-1-85 ~]$ ping www.amazon.com
PING www-amazon-com.customer.fastly.net (18.65.233.187) 56(84) bytes of data.
64 bytes from 18.65.233.187 (18.65.233.187): icmp_seq=15 ttl=58 time=2.06 ms
64 bytes from 18.65.233.187 (18.65.233.187): icmp_seq=16 ttl=58 time=2.26 ms
```

请注意，运行 ping 命令成功并且流量未被阻止。

3. 使用公有 IPv4 地址对实例 B 运行 Ping 命令以检查入站流量：

```
ping 3.18.106.198
```

输出：

```
Pinging 3.18.106.198 with 32 bytes of data:
Reply from 3.18.106.198: bytes=32 time=83ms TTL=110
Reply from 3.18.106.198: bytes=32 time=54ms TTL=110
```

请注意，运行 ping 命令成功并且流量未被阻止。

4. 使用私有 IPv4 地址连接并检查出站流量：

```
aws ec2-instance-connect ssh --instance-id  i-08552a0774b5c8f72 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.
Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #   ~_  ####        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
```

```
~~~         /
~~..   _/
/ /
/m/'
Last login: Fri Sep 27 18:12:27 2024 from 3.16.146.5
[ec2-user@ip-10-0-2-98 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
64 bytes from server-3-160-24-126.cmh68.r.cloudfront.net (18.65.233.187):
 icmp_seq=1 ttl=249 time=1.55 ms
64 bytes from server-3-160-24-126.cmh68.r.cloudfront.net (18.65.233.187):
 icmp_seq=2 ttl=249 time=1.67 ms
```

请注意，运行 ping 命令成功并且流量未被阻止。

5. 连接到实例 C。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-04eca55f2a482b2c4 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.
Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #   ~_  ####        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~..   _/
/ /
/m/'
Last login: Thu Sep 19 20:31:26 2024 from 10.0.2.86
[ec2-user@ip-10-0-3-180 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
64 bytes from server-3-160-24-126.cmh68.r.cloudfront.net (18.65.233.187):
 icmp_seq=1 ttl=248 time=1.75 ms
64 bytes from server-3-160-24-126.cmh68.r.cloudfront.net (18.65.233.187):
 icmp_seq=2 ttl=248 time=1.97 ms
64 bytes from server-3-160-24-26.cmh68.r.cloudfront.net (18.65.233.187):
 icmp_seq=3 ttl=248 time=1.08 ms
```

请注意，运行 ping 命令成功并且流量未被阻止。

6. 连接到实例 D。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-05f9e6a9cfac1dba0 --region us-
east-2 --connection-type eice
```

输出：

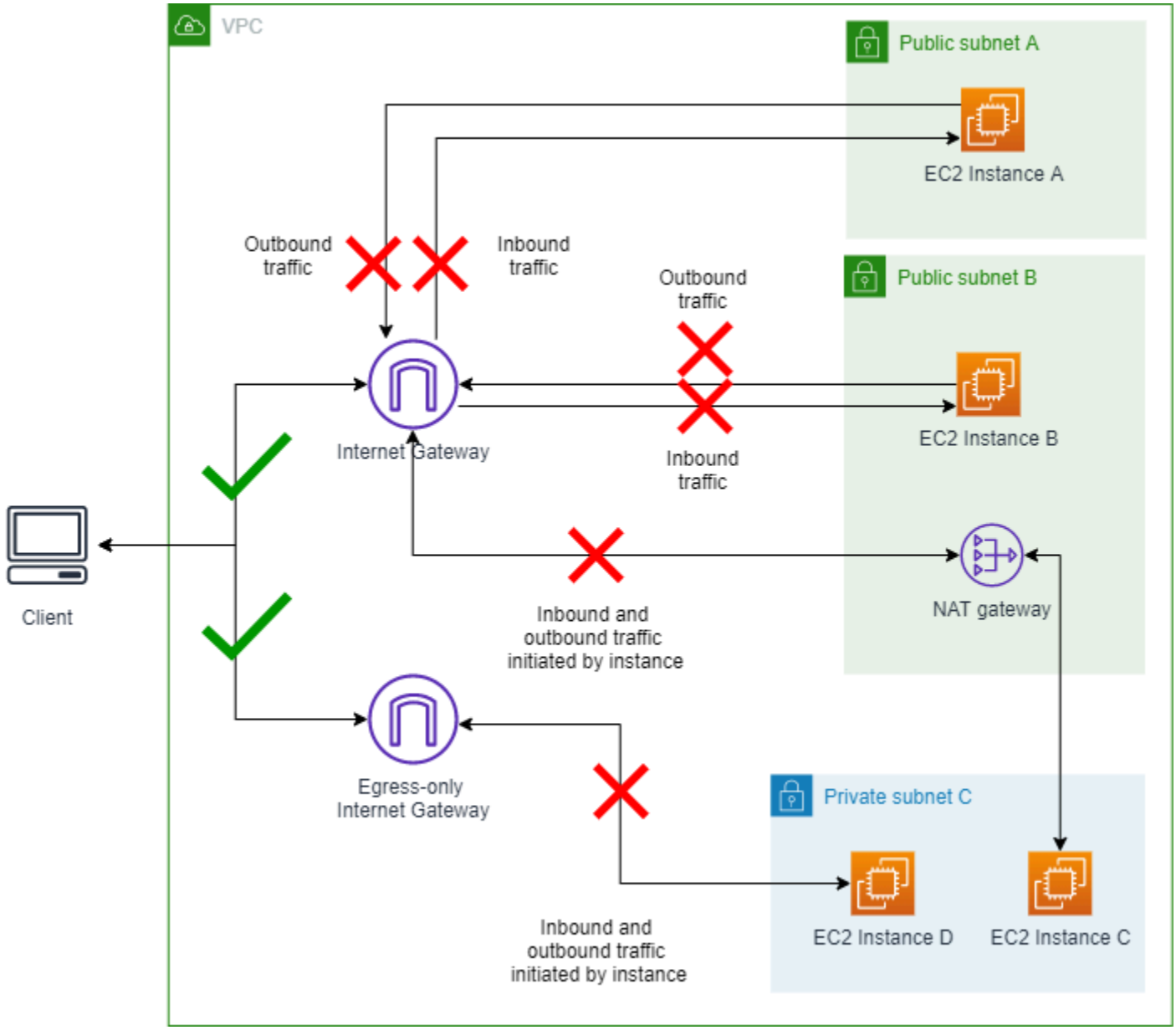
```
The authenticity of host '10.0.3.59 can't be established.
ECDSA key fingerprint is SHA256:c4naBCqbC61/cExDyccEproNU+1HHSpMSzl2J6cOtIZA8g.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.3.59' (ECDSA) to the list of known hosts.
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #   ~_  ####        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~..   _/
_/ _/
_/m/'
[ec2-user@ip-10-0-3-59 ~]$ ping www.amazon.com
PING www.amazon.com(2600:9000:25f3:ee00:7:49a5:5fd4:b121
 (2600:9000:25f3:ee00:7:49a5:5fd4:b121)) 56 data bytes
64 bytes from 2600:9000:25f3:ee00:7:49a5:5fd4:b121
 (2600:9000:25f3:ee00:7:49a5:5fd4:b121): icmp_seq=1 ttl=58 time=1.19 ms
64 bytes from 2600:9000:25f3:ee00:7:49a5:5fd4:b121
 (2600:9000:25f3:ee00:7:49a5:5fd4:b121): icmp_seq=2 ttl=58 time=1.38 ms
```

请注意，运行 ping 命令成功并且流量未被阻止。

场景 2 - 启用 VPC BPA 双向模式

在本节中，您将启用 VPC BPA，并在阻止进出您账户中互联网网关的流量。

显示 VPC BPA 双向模式启用的示意图：



2.1 启用 VPC BPA 双向模式

完成本节以启用 VPC BPA。VPC BPA 双向模式阻止进出此区域互联网网关和仅出口互联网网关的所有流量（已排除 VPC 和子网除外）。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在左侧导航窗格中，选择设置。
3. 选择编辑公共访问权限设置。

4. 选择开启屏蔽公共访问和双向，然后选择保存更改。
5. 等待状态更改为开启。VPC BPA 设置生效和状态更新可能需要几分钟时间。

VPC BPA 现已启用。

AWS CLI

1. 使用 modify-vpc-block-public-access-options 命令启用 VPC BPA：

```
aws ec2 --region us-east-2 modify-vpc-block-public-access-options --internet-
gateway-block-mode block-bidirectional
```

VPC BPA 设置生效和状态更新可能需要几分钟时间。

2. 查看 VPC BPA 的状态：

```
aws ec2 --region us-east-2 describe-vpc-block-public-access-options
```

2.2 连接到实例

完成本节以连接到您的实例。

AWS Management Console

1. 如同在方案 1 中所做的那样，对实例 A 和实例 B 的公有 IPv4 地址运行 Ping 命令。请注意，流量已被阻止。
2. 按照场景 1 中的方式，通过 EC2 Instance Connect > 使用 EC2 Instance Connect 端点连接选项连接到实例 A。确保您使用端点选项。
3. 选择连接。成功连接到实例后，ping www.amazon.com。请注意，所有出站流量均已阻止。
4. 使用与连接实例 A 相同的方法连接到实例 B、C 和 D，测试对互联网的出站请求。请注意，所有出站流量均已阻止。

AWS CLI

1. 使用公有 IPv4 地址对实例 A 运行 Ping 命令以检查入站流量：

```
ping 18.225.8.244
```

输出：

```
Pinging 18.225.8.244 with 32 bytes of data:

Request timed out.
```

请注意，运行 ping 命令失败并且流量将被阻止。

2. 使用私有 IPv4 地址连接并检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-058db34f9a0997895 --region us-
east-2 --connection-type eice
```

输出：

```
The authenticity of host '10.0.1.85' can't be established.
ECDSA key fingerprint is SHA256:3zo/gSss+HAZ+7eTyWlOB/Ke04IM+hadjsoLJeRTWBk.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.1.85' (ECDSA) to the list of known hosts.
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #_   ~_  ####_        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~._.   _/
/ /
/m/'
Last login: Fri Sep 27 14:16:53 2024 from 3.16.146.5
[ec2-user@ip-10-0-1-85 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
```

请注意，运行 ping 命令失败并且流量将被阻止。

3. 使用公有 IPv4 地址对实例 B 运行 Ping 命令以检查入站流量：

```
ping 3.18.106.198
```

输出：

```
Pinging 3.18.106.198 with 32 bytes of data:
Request timed out.
```

请注意，运行 ping 命令失败并且流量将被阻止。

4. 使用私有 IPv4 地址连接并检查出站流量：

```
aws ec2-instance-connect ssh --instance-id  i-08552a0774b5c8f72 --region us-
east-2 --connection-type eice
```

输出：

```
The authenticity of host '10.0.2.98' can't be established.
ECDSA key fingerprint is SHA256:0IjXKKyVlDthcCfI0IPIJMUiItAOLYKRNLGTYURnFXo.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.2.98' (ECDSA) to the list of known hosts.
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #   ~_  ####        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~..   _/
/ /
/m/'
Last login: Fri Sep 27 14:18:16 2024 from 3.16.146.5
[ec2-user@ip-10-0-2-98 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
```

请注意，运行 ping 命令失败并且流量将被阻止。

5. 连接到实例 C。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-04eca55f2a482b2c4 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #   ~_  ####        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~..   _/
/ /
/m/'
Last login: Tue Sep 24 15:17:56 2024 from 10.0.2.86
[ec2-user@ip-10-0-3-180 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
```

请注意，运行 ping 命令失败并且流量将被阻止。

6. 连接到实例 D。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-05f9e6a9cfac1dba0 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #   ~_  ####        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~..   _/
_/ _/
_/m/'
Last login: Fri Sep 27 16:42:01 2024 from 3.16.146.5
[ec2-user@ip-10-0-3-59 ~]$ ping www.amazon.com
PING www.amazon.com(2600:9000:25f3:8200:7:49a5:5fd4:b121
 (2600:9000:25f3:8200:7:49a5:5fd4:b121)) 56 data bytes
```

请注意，运行 ping 命令失败并且流量将被阻止。

2.3 可选：使用 Reachability Analyzer 验证连接是否被阻止

VPC Reachability Analyzer 可用于了解根据您的网络配置（包括 VPC BPA 设置）是否可以访问某些网络路径。在本示例中，您将分析之前尝试的相同网络路径，以确认 VPC BPA 是连接失败的原因。

AWS Management Console

1. 转到位于 https://console.aws.amazon.com/networkinsights/home#ReachabilityAnalyzer 的网络见解控制台。
2. 单击创建和分析路径。
3. 对于源类型，选择互联网网关，然后从源下拉列表中选择标记为 VPC BPA 互联网网关的互联网网关。
4. 对于目标类型，选择实例，然后从目标下拉列表中选择标记为 VPC BPA 实例 A 的实例。
5. 单击创建和分析路径。
6. 等待分析完成。这可能需要几分钟时间。
7. 完成后，您应该会看到可访问性状态为无法访问，并且路径详细信息显示原因是 VPC_BLOCK_PUBLIC_ACCESS_ENABLED。

AWS CLI

1. 使用标记为 VPC BPA 互联网网关的互联网网关的 ID 和标记为 VPC BPA 实例 A 的实例 ID 创建网络路径：

```
aws ec2 --region us-east-2 create-network-insights-path --source igw-id --
destination instance-id --protocol TCP
```

2. 开始对网络路径进行分析：

```
aws ec2 --region us-east-2 start-network-insights-analysis --network-insights-
path-id nip-id
```

3. 检索分析的结果：

```
aws ec2 --region us-east-2 describe-network-insights-analyses --network-
insights-analysis-ids nia-id
```

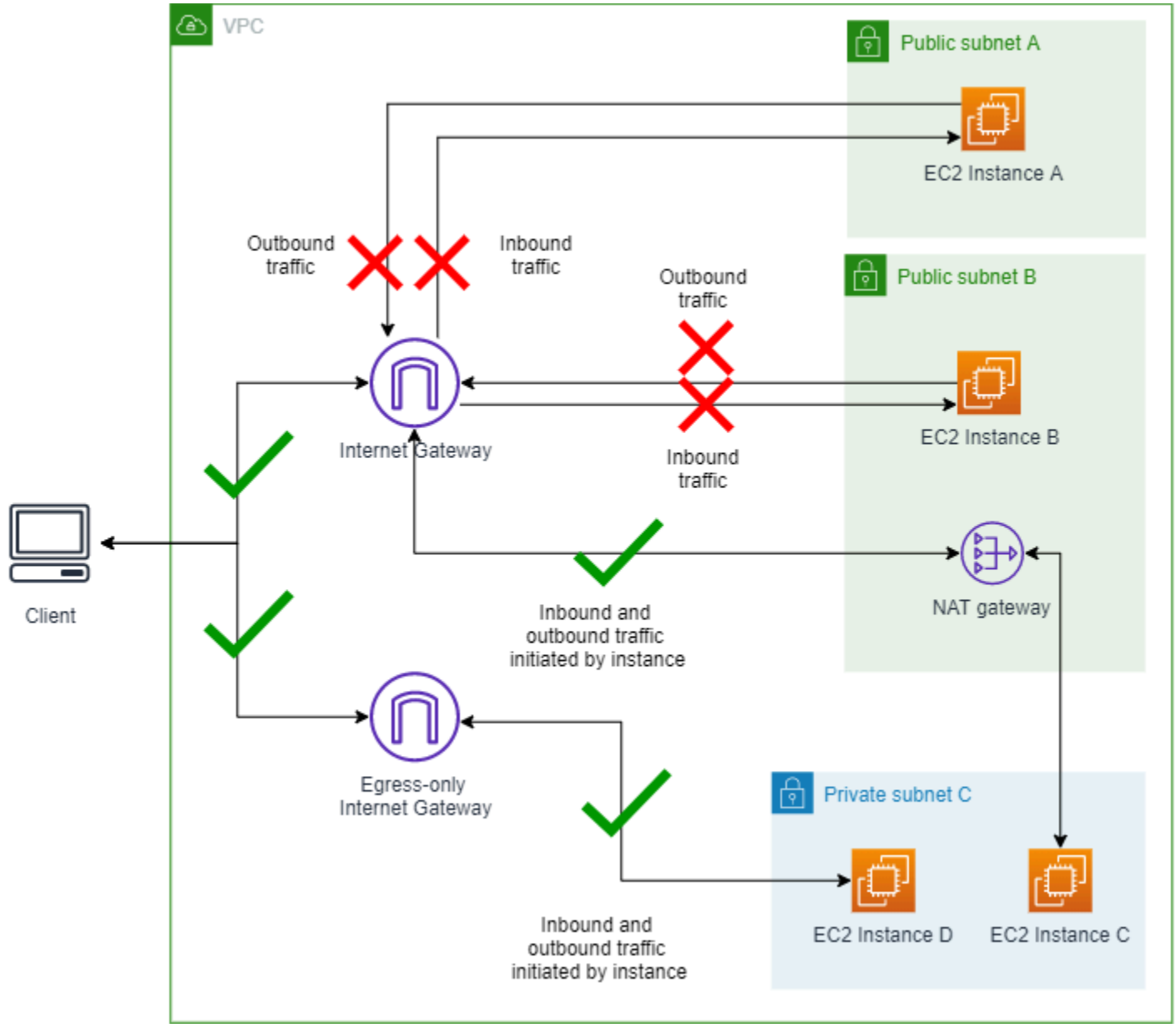
4. 请确认 VPC_BLOCK_PUBLIC_ACCESS_ENABLED 是无法访问的 ExplanationCode。

请注意，您也可以使用流日志监控 VPC BPA 影响。

场景 3 - 将 VPC BPA 更改为仅入口模式

在本节中，您将更改 VPC BPA 流量方向，只允许使用 NAT 网关或仅出口互联网网关的流量。公有子网中的 EC2 实例 A 和 B 无法通过互联网访问，因为 BPA 会阻止通过互联网网关的入站流量。私有子网中的实例 C 和实例 D 仍能够通过 NAT 网关和仅出口互联网网关发出出站流量，因此仍然可以访问互联网。

VPC BPA 仅入口模式已启用的示意图：



3.1 将模式更改为仅入口

完成本节以更改模式。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在左侧导航窗格中，选择设置。
3. 在阻止公开访问选项卡中，选择编辑公共访问权限设置。
4. 在 VPC 控制台中修改公共访问设置，并将方向更改为仅入口。
5. 保存更改并等待状态更新。VPC BPA 设置生效和状态更新可能需要几分钟时间。

AWS CLI

1. 修改 VPC BPA 模式：

   ```
   aws ec2 --region us-east-2 modify-vpc-block-public-access-options --internet-gateway-block-mode block-ingress
   ```

   VPC BPA 设置生效和状态更新可能需要几分钟时间。
2. 查看 VPC BPA 的状态：

   ```
   aws ec2 --region us-east-2 describe-vpc-block-public-access-options
   ```

3.2 连接到实例

完成本节以连接到实例。

AWS Management Console

1. 如同在方案 1 中所做的那样，对实例 A 和实例 B 的公有 IPv4 地址运行 Ping 命令。请注意，流量已被阻止。
2. 如同在方案 1 中所做的那样，使用 EC2 Instance Connect 连接到实例 A 和实例 B，然后从这些实例运行 ping www.amazon.com 命令。请注意，您无法从实例 A 或实例 B 对互联网上的公共站点运行 ping 命令并且流量已被阻止。

3. 如同在方案 1 中所做的那样，使用 EC2 Instance Connect 连接到实例 C 和实例 D，然后从这些实例运行 ping www.amazon.com 命令。请注意，您可以从实例 C 或实例 D 对互联网上的公共站点运行 ping 命令并且允许流量。

AWS CLI

1. 使用公有 IPv4 地址对实例 A 运行 Ping 命令以检查入站流量：

```
ping 18.225.8.244
```

输出：

```
Pinging 18.225.8.244 with 32 bytes of data:

Request timed out.
```

请注意，运行 ping 命令失败并且流量将被阻止。

2. 使用私有 IPv4 地址连接并检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-058db34f9a0997895 --region us-
east-2 --connection-type eice
```

输出：

```
The authenticity of host '10.0.1.85' can't be established.
ECDSA key fingerprint is SHA256:3zo/gSss+HAZ+7eTyWlOB/Ke04IM+hadjsoLJeRTWBk.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.1.85' (ECDSA) to the list of known hosts.
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #_   ~_  ####_        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~._.   _/
/ /
/m/'
Last login: Fri Sep 27 14:16:53 2024 from 3.16.146.5
[ec2-user@ip-10-0-1-85 ~]$ ping www.amazon.com
```

```
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
```

请注意，运行 ping 命令失败并且流量将被阻止。

3. 使用公有 IPv4 地址对实例 B 运行 Ping 命令以检查入站流量：

```
ping 3.18.106.198
```

输出：

```
Pinging 3.18.106.198 with 32 bytes of data:
Request timed out.
```

请注意，运行 ping 命令失败并且流量将被阻止。

4. 使用私有 IPv4 地址连接并检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-08552a0774b5c8f72 --region us-
east-2 --connection-type eice
```

输出：

```
The authenticity of host '10.0.2.98 ' can't be established.
ECDSA key fingerprint is SHA256:0IjXKKyVlDthcCfI0IPIJMUiItAOLYKRNLGTYURnFXo.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.2.98' (ECDSA) to the list of known hosts.
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #   ~_  ####        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~..   _/
_/ /
/m/'
Last login: Fri Sep 27 14:18:16 2024 from 3.16.146.5
[ec2-user@ip-10-0-2-98 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
```

请注意，运行 ping 命令失败并且流量将被阻止。

5. 连接到实例 C。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-04eca55f2a482b2c4 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'

Last login: Tue Sep 24 15:28:09 2024 from 10.0.2.86

[ec2-user@ip-10-0-3-180 ~]$ ping www.amazon.com

PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.

64 bytes from server-3-160-24-126.cmh68.r.cloudfront.net (18.65.233.187):
 icmp_seq=1 ttl=248 time=1.84 ms
64 bytes from server-3-160-24-126.cmh68.r.cloudfront.net (18.65.233.187):
 icmp_seq=2 ttl=248 time=1.40 ms
```

请注意，运行 ping 命令成功并且流量未被阻止。

6. 连接到实例 D。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-05f9e6a9cfac1dba0 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
```

```
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Sep 27 16:48:38 2024 from 3.16.146.5
[ec2-user@ip-10-0-3-59 ~]$ ping www.amazon.com
PING www.amazon.com(2600:9000:25f3:5800:7:49a5:5fd4:b121
 (2600:9000:25f3:5800:7:49a5:5fd4:b121)) 56 data bytes
64 bytes from 2600:9000:25f3:5800:7:49a5:5fd4:b121
 (2600:9000:25f3:5800:7:49a5:5fd4:b121): icmp_seq=14 ttl=58 time=1.47 ms
64 bytes from 2600:9000:25f3:5800:7:49a5:5fd4:b121
 (2600:9000:25f3:5800:7:49a5:5fd4:b121): icmp_seq=16 ttl=58 time=1.59 ms
```

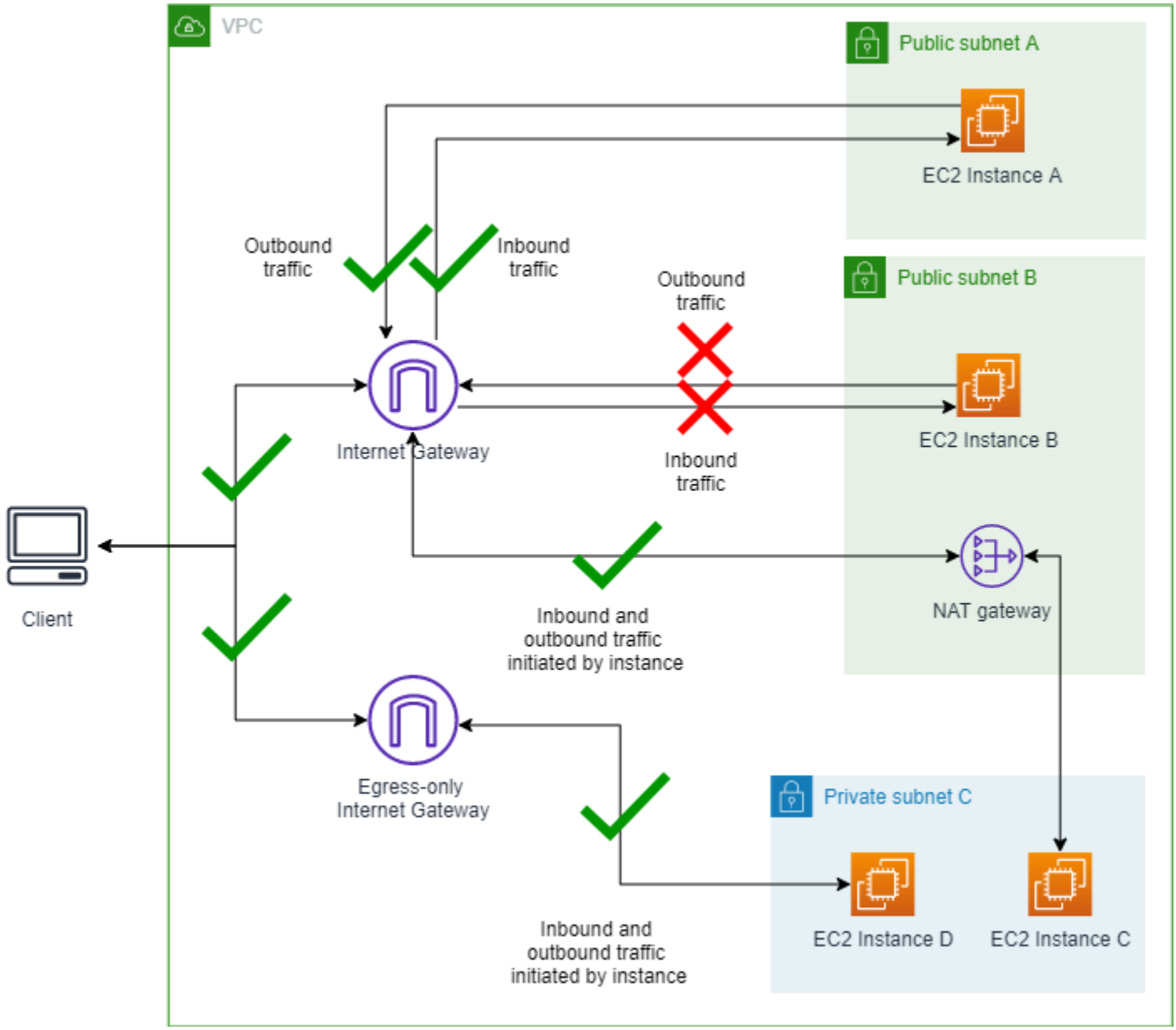
请注意，运行 ping 命令成功并且流量未被阻止。

场景 4 - 创建排除项

在本部分中，您将创建一个排除项。然后，VPC BPA 将只阻止没有排除项的子网上的流量。VPC BPA 排除是一种可以应用于单个 VPC 或子网的模式，可将其排除在账户的 VPC BPA 模式之外，并允许双向或仅出口访问。即使账户未启用 VPC BPA，您也可以为 VPC 和子网创建 VPC BPA 排除项，以确保启用 VPC BPA 时排除项不会中断流量。

在本示例中，我们将为子网 A 创建排除项，以显示 VPC BPA 如何影响排除项的流量。

VPC BPA 仅入口模式已启用且子网 A 排除项已启用双向模式的示意图：



4.1 为子网 A 创建排除项

完成本节以创建排除项。VPC BPA 排除是一种可以应用于单个 VPC 或子网的模式，可将其排除在账户的 VPC BPA 模式之外，并允许双向或仅出口访问。即使账户未启用 VPC BPA，您也可以为 VPC 和子网创建 VPC BPA 排除项，以确保启用 VPC BPA 时排除项不会中断流量。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在左侧导航窗格中，选择设置。
3. 在屏蔽公共访问选项卡的排除下，选择创建排除项。

4. 选择 VPC BPA 公有子网 A，确保选择的允许方向为双向，然后选择创建排除项。
5. 等待排除项状态变为活动。您可能需要刷新排除项表才能查看更改。

排除项已创建。

AWS CLI

1. 修改排除允许方向：

```
aws ec2 --region us-east-2 create-vpc-block-public-access-exclusion --subnet-
id subnet-id --internet-gateway-exclusion-mode allow-bidirectional
```

2. 更新排除项状态可能需要一段时间。要查看排除项的状态：

```
aws ec2 --region us-east-2 describe-vpc-block-public-access-exclusions --
exclusion-ids exclusion-id
```

4.2 连接到实例

完成本节以连接到实例。

AWS Management Console

1. 对实例 A 的公有 IPv4 地址运行 Ping 命令。请注意，允许流量。
2. 对实例 B 的公有 IPv4 地址运行 Ping 命令。请注意，流量已被阻止。
3. 如同在方案 1 中所做的那样，使用 EC2 Instance Connect 连接到实例 A，然后运行 ping www.amazon.com 命令。请注意，您可以从实例 A 对互联网上的公共站点运行 ping 命令。允许流量。
4. 如同在方案 1 中所做的那样，使用 EC2 Instance Connect 连接到实例 B，然后从该实例运行 ping www.amazon.com 命令。请注意，您无法从实例 B 对互联网上的公共站点运行 ping 命令。流量已被阻止。
5. 如同在方案 1 中所做的那样，使用 EC2 Instance Connect 连接到实例 C 和实例 D，然后从这些实例运行 ping www.amazon.com 命令。请注意，您可以从实例 C 或实例 D 对互联网上的公共站点运行 ping 命令。允许流量。

AWS CLI

1. 使用公有 IPv4 地址对实例 A 运行 Ping 命令以检查入站流量：

```
ping 18.225.8.244
```

输出：

```
Pinging 18.225.8.244 with 32 bytes of data:

Reply from 18.225.8.244: bytes=32 time=51ms TTL=110
Reply from 18.225.8.244: bytes=32 time=61ms TTL=110
```

请注意，运行 ping 命令成功并且流量未被阻止。

2. 使用私有 IPv4 地址连接并检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-058db34f9a0997895 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #_   ~_  ####_        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~._.   _/
/ /
/m/'
Last login: Fri Sep 27 17:58:12 2024 from 3.16.146.5
[ec2-user@ip-10-0-1-85 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
64 bytes from server-3-160-24-126.cmh68.r.cloudfront.net (18.65.233.187):
 icmp_seq=1 ttl=249 time=1.03 ms
64 bytes from server-3-160-24-126.cmh68.r.cloudfront.net (18.65.233.187):
 icmp_seq=2 ttl=249 time=1.72 ms
```

请注意，运行 ping 命令成功并且流量未被阻止。

3. 使用公有 IPv4 地址对实例 B 运行 Ping 命令以检查入站流量：

```
ping 3.18.106.198
```

输出：

```
Pinging 3.18.106.198 with 32 bytes of data:
Request timed out.
```

请注意，运行 ping 命令失败并且流量将被阻止。

4. 使用私有 IPv4 地址连接并检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-08552a0774b5c8f72 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #   ~_  ####        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~..   _/
_/ /
/m/'
Last login: Fri Sep 27 18:12:03 2024 from 3.16.146.5
[ec2-user@ip-10-0-2-98 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
```

请注意，运行 ping 命令失败并且流量将被阻止。

5. 连接到实例 C。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-04eca55f2a482b2c4 --region us-
east-2 --connection-type eice
```

输出

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
,     #   ~_  ####        Amazon Linux 2023
~~  _#####\  ~~     ###|
~~       #/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '->
~~~         /
~~..   _/
_/ /
/m/'

Last login: Tue Sep 24 15:28:09 2024 from 10.0.2.86

[ec2-user@ip-10-0-3-180 ~]$ ping www.amazon.com

PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.

64 bytes from server-3-160-24-126.cmh68.r.cloudfront.net (18.65.233.187):
 icmp_seq=1 ttl=248 time=1.84 ms
64 bytes from server-3-160-24-126.cmh68.r.cloudfront.net (18.65.233.187):
 icmp_seq=2 ttl=248 time=1.40 ms
```

请注意，运行 ping 命令成功并且流量未被阻止。

6. 连接到实例 D。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-05f9e6a9cfac1dba0 --region us-
east-2 --connection-type eice
```

输出

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
```

```
       _/m/'
Last login: Fri Sep 27 18:00:52 2024 from 3.16.146.5
[ec2-user@ip-10-0-3-59 ~]$ ping www.amazon.com
PING
 www.amazon.com(g2600-141f-4000-059a-0000-0000-0000-3bd4.deploy.static.akamaitechnologie
 (2600:141f:4000:59a::3bd4)) 56 data bytes
64 bytes from
 g2600-141f-4000-059a-0000-0000-0000-3bd4.deploy.static.akamaitechnologies.com
 (2600:141f:4000:59a::3bd4): icmp_seq=1 ttl=48 time=15.9 ms
64 bytes from
 g2600-141f-4000-059a-0000-0000-0000-3bd4.deploy.static.akamaitechnologies.com
 (2600:141f:4000:59a::3bd4): icmp_seq=2 ttl=48 time=15.8 ms
```

请注意，运行 ping 命令成功并且流量未被阻止。

4.3 可选：使用 Reachability Analyzer 验证连接

使用与方案 2 在 Reachability Analyzer 中创建的相同网络路径，由于已经为公有子网 A 创建了排除项，您现在可以运行新分析并确认路径目前是否可访问。

有关 Reachability Analyzer 区域可用性的信息，请参阅《Reachability Analyzer Guide》中的 Considerations。

AWS Management Console

1. 从您之前在网络见解控制台中创建的网络路径，单击重新运行分析。
2. 等待分析完成。该过程可能需要几分钟。
3. 确认该路径现在可访问。

AWS CLI

1. 使用之前创建的网络路径 ID 开始新分析：

```
aws ec2 --region us-east-2 start-network-insights-analysis --network-insights-
path-id nip-id
```

2. 检索分析的结果：

```
aws ec2 --region us-east-2 describe-network-insights-analyses --network-
insights-analysis-ids nia-id
```

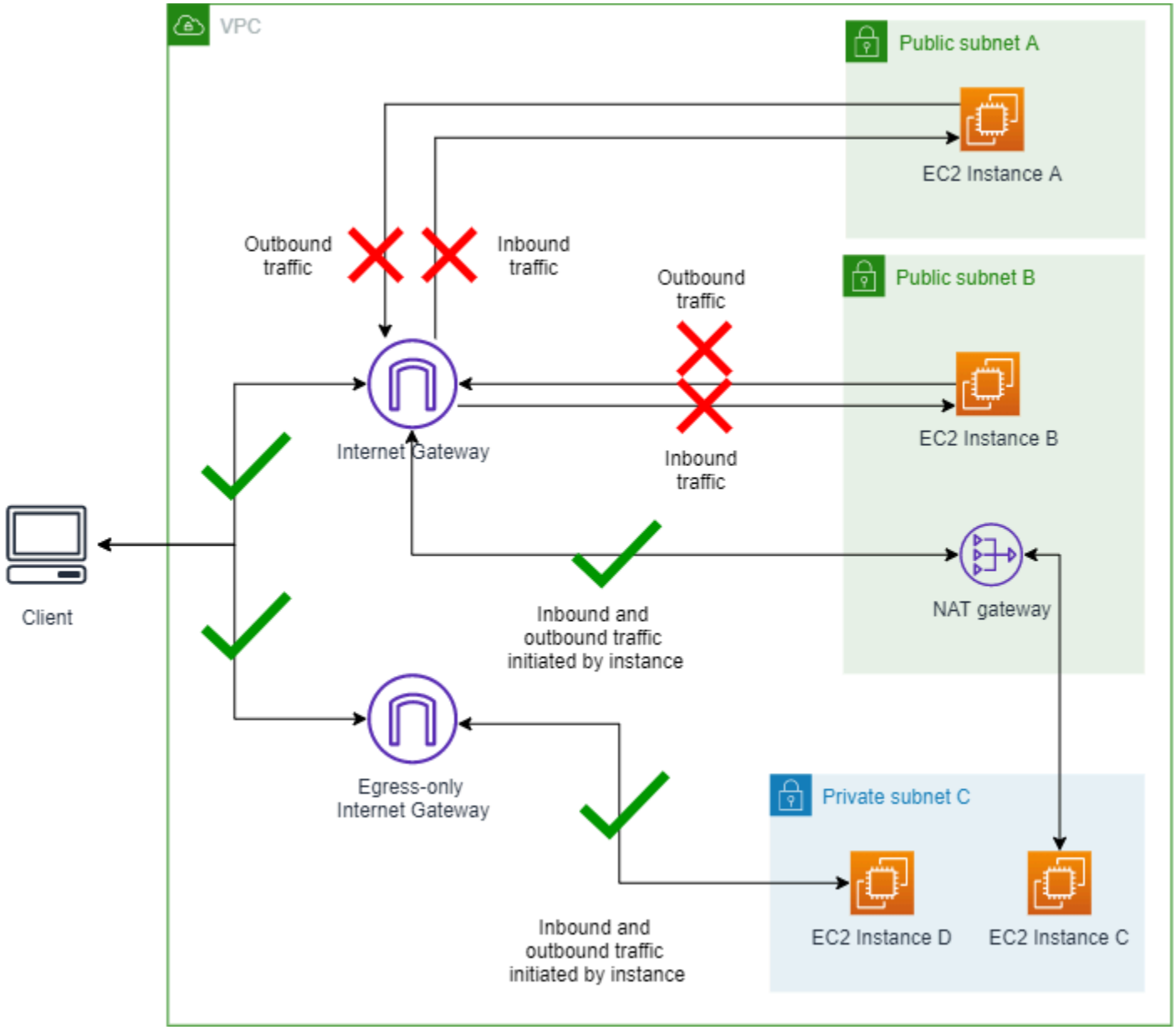
3. 确认 VPC_BLOCK_PUBLIC_ACCESS_ENABLED 说明代码不再存在。

场景 5 - 修改排除模式

在本节中，您将更改排除项的允许流量方向，以了解其如何影响 VPC BPA。

> **Note**
>
> 在这一场景下，您需要将排除模式更改为仅出口。请注意在这样操作时，子网 A 上的“仅出口”排除项不允许出站流量，这是有违直觉的，因为您原本希望它允许出站流量。但是，由于账户级 BPA 为仅入口模式，因此仅出口排除项将被忽略，并且子网 A 到互联网网关的路由受到 VPC BPA 的限制，从而阻止了出站流量。要在子网 A 上启用出站流量，就必须将 VPC BPA 切换到双向模式。

VPC BPA 仅入口模式已启用且子网 A 排除项已启用仅出口模式的示意图：



5.1 将排除项允许方向更改为仅出口

完成本节以更改排除项允许方向。

AWS Management Console

1. 编辑您在方案 4 中创建的排除项，并将允许方向更改为仅出口。
2. 选择保存更改。
3. 等待排除项状态变为活动。VPC BPA 设置生效和状态更新可能需要几分钟时间。您可能需要刷新排除项表才能查看更改。

AWS CLI

1. 修改排除允许方向：

```
aws ec2 --region us-east-2 modify-vpc-block-public-access-exclusion --exclusion-
id exclusion-id --internet-gateway-exclusion-mode allow-egress
```

   VPC BPA 设置生效和状态更新可能需要几分钟时间。

2. 更新排除项状态可能需要一段时间。要查看排除项的状态：

```
aws ec2 --region us-east-2 describe-vpc-block-public-access-exclusion
```

5.2 连接到实例

完成本节以连接到实例。

AWS Management Console

1. 对实例 A 和实例 B 的公有 IPv4 地址运行 Ping 命令。请注意，流量已被阻止。
2. 如同在方案 1 中所做的那样，使用 EC2 Instance Connect 连接到实例 A 和实例 B，然后运行 ping www.amazon.com 命令。请注意，您无法从实例 A 或实例 B 对互联网上的公共站点运行 ping 命令。流量已被阻止。
3. 如同在方案 1 中所做的那样，使用 EC2 Instance Connect 连接到实例 C 和实例 D，然后从这些实例运行 ping www.amazon.com 命令。请注意，您可以从实例 C 或实例 D 对互联网上的公共站点运行 ping 命令。允许流量。

AWS CLI

1. 使用公有 IPv4 地址对实例 A 运行 Ping 命令以检查入站流量：

```
ping 18.225.8.244
```

   输出：

```
Pinging 18.225.8.244 with 32 bytes of data:
Request timed out.
```

请注意，运行 ping 命令失败并且流量将被阻止。

2. 使用私有 IPv4 地址连接并检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-058db34f9a0997895 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Sep 27 18:09:55 2024 from 3.16.146.5
[ec2-user@ip-10-0-1-85 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
```

请注意，运行 ping 命令失败并且流量将被阻止。

3. 使用公有 IPv4 地址对实例 B 运行 Ping 命令以检查入站流量：

```
ping 3.18.106.198
```

输出：

```
Pinging 3.18.106.198 with 32 bytes of data:
Request timed out.
```

请注意，运行 ping 命令失败并且流量将被阻止。

4. 使用私有 IPv4 地址连接并检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-058db34f9a0997895 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Sep 27 18:09:55 2024 from 3.16.146.5
[ec2-user@ip-10-0-1-85 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
```

请注意，运行 ping 命令失败并且流量将被阻止。

5. 连接到实例 C。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-04eca55f2a482b2c4 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Sep 27 18:00:31 2024 from 3.16.146.5
[ec2-user@ip-10-0-3-180 ~]$ ping www.amazon.com
PING www.amazon.com(2600:9000:25f3:a600:7:49a5:5fd4:b121
 (2600:9000:25f3:a600:7:49a5:5fd4:b121)) 56 data bytes
```

```
64 bytes from 2600:9000:25f3:a600:7:49a5:5fd4:b121
 (2600:9000:25f3:a600:7:49a5:5fd4:b121): icmp_seq=1 ttl=58 time=1.51 ms
64 bytes from 2600:9000:25f3:a600:7:49a5:5fd4:b121
 (2600:9000:25f3:a600:7:49a5:5fd4:b121): icmp_seq=2 ttl=58 time=1.49 ms
```

请注意，运行 ping 命令成功并且流量未被阻止。

6. 连接到实例 D。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-05f9e6a9cfac1dba0 --region us-
east-2 --connection-type eice
```

输出：

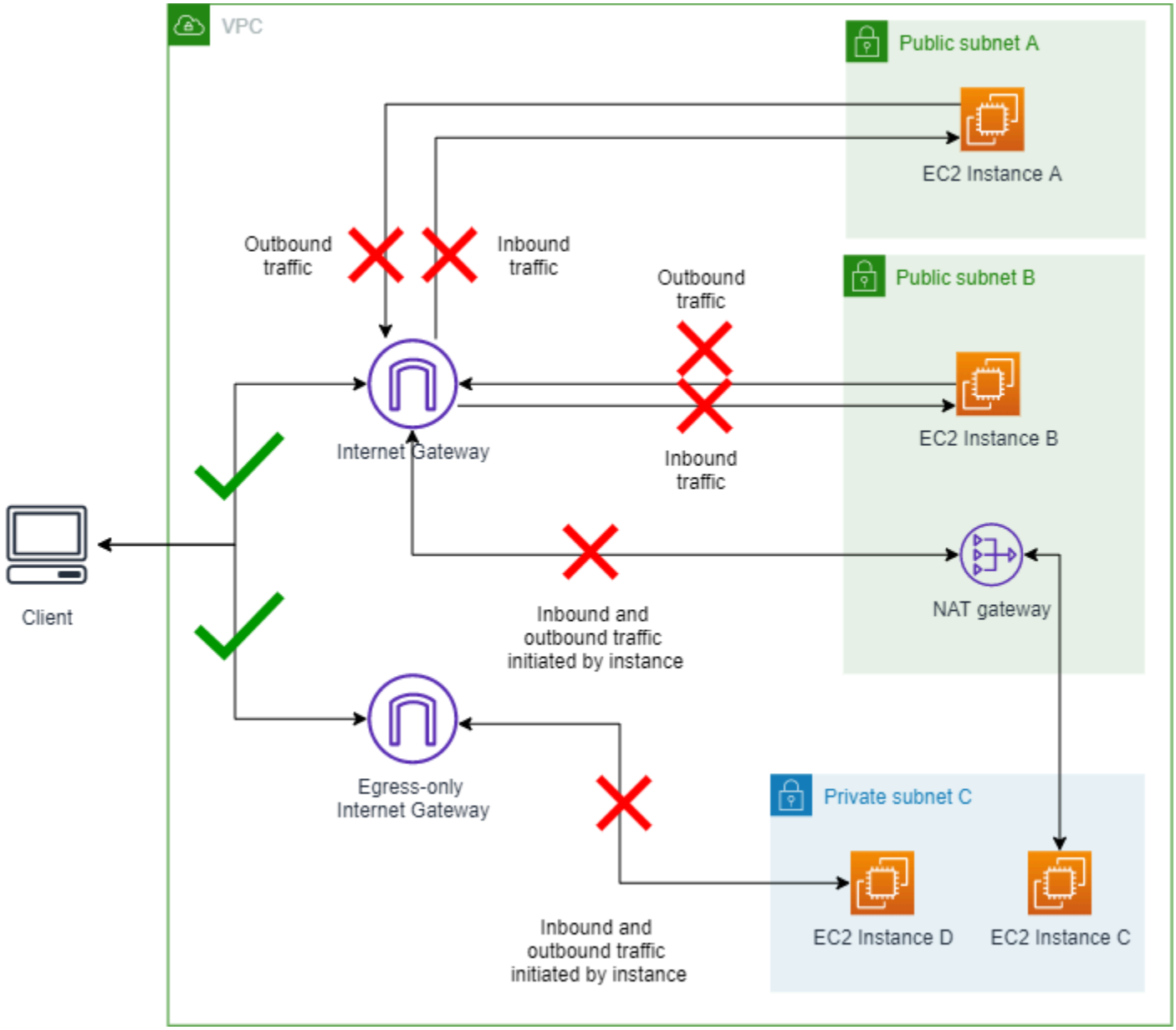
```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Sep 27 18:13:55 2024 from 3.16.146.5
[ec2-user@ip-10-0-3-59 ~]$ ping www.amazon.com
PING www.amazon.com(2606:2cc0::374 (2606:2cc0::374)) 56 data bytes
64 bytes from 2606:2cc0::374 (2606:2cc0::374): icmp_seq=1 ttl=58 time=1.21 ms
64 bytes from 2606:2cc0::374 (2606:2cc0::374): icmp_seq=2 ttl=58 time=1.51 ms
```

请注意，运行 ping 命令成功并且流量未被阻止。

场景 6 - 修改 VPC BPA 模式

在本节中，您将更改 VPC BPA 阻止方向，以了解其如何影响流量。在本方案中，双向模式下启用的 VPC BPA 会阻止所有流量，就像在方案 1 中一样。除非排除项可以访问 NAT 网关或仅出口互联网网关，否则流量将被阻止。

VPC BPA 双向模式已启用且子网 A 排除项已启用仅入口模式的示意图：



6.1 将 VPC BPA 更改为双向模式

完成本节以更改 VPC BPA 模式。

AWS Management Console

1. 通过 https://console.aws.amazon.com/vpc/ 打开 Amazon VPC 控制台。
2. 在左侧导航窗格中，选择设置。
3. 选择编辑公共访问权限设置。
4. 将阻止方向更改为双向，然后选择保存更改。

5. 等待状态更改为开启。VPC BPA 设置生效和状态更新可能需要几分钟时间。

AWS CLI

1. 修改 VPC BPA 阻止方向：

```
aws ec2 --region us-east-2 modify-vpc-block-public-access-options --internet-
gateway-block-mode block-bidirectional
```

   VPC BPA 设置生效和状态更新可能需要几分钟时间。

2. 查看 VPC BPA 的状态：

```
aws ec2 --region us-east-2 describe-vpc-block-public-access-options
```

6.2 连接到实例

完成本节以连接到实例。

AWS Management Console

1. 对实例 A 和实例 B 的公有 IPv4 地址运行 Ping 命令。请注意，流量已被阻止。
2. 如同在方案 1 中所做的那样，使用 EC2 Instance Connect 连接到实例 A 和实例 B，然后运行 ping www.amazon.com 命令。请注意，您无法从实例 A 或实例 B 对互联网上的公共站点运行 ping 命令。流量已被阻止。
3. 如同在方案 1 中所做的那样，使用 EC2 Instance Connect 连接到实例 C 和实例 D，然后从这些实例运行 ping www.amazon.com 命令。请注意，您无法从实例 C 或实例 D 对互联网上的公共站点运行 ping 命令。流量已被阻止。

AWS CLI

1. 使用公有 IPv4 地址对实例 A 运行 Ping 命令以检查入站流量：

```
ping 18.225.8.244
```

   输出：

```
Pinging 18.225.8.244 with 32 bytes of data:
```

```
Request timed out.
```

请注意，运行 ping 命令失败并且流量将被阻止。

2. 使用私有 IPv4 地址连接并检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-058db34f9a0997895 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Sep 27 18:17:44 2024 from 3.16.146.5
[ec2-user@ip-10-0-1-85 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
```

请注意，运行 ping 命令失败并且流量将被阻止。

3. 使用公有 IPv4 地址对实例 A 运行 Ping 命令以检查入站流量：

```
ping 3.18.106.198
```

输出：

```
Pinging 3.18.106.198 with 32 bytes of data:
Request timed out.
```

请注意，运行 ping 命令失败并且流量将被阻止。

4. 使用私有 IPv4 地址连接并检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-058db34f9a0997895 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Sep 27 18:09:55 2024 from 3.16.146.5
[ec2-user@ip-10-0-1-85 ~]$ ping www.amazon.com
PING d3ag4hukkh62yn.cloudfront.net (18.65.233.187) 56(84) bytes of data.
```

请注意，运行 ping 命令失败并且流量将被阻止。

5. 连接到实例 C。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-04eca55f2a482b2c4 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Sep 27 18:19:45 2024 from 3.16.146.5
[ec2-user@ip-10-0-3-180 ~]$ ping www.amazon.com
```

```
PING www.amazon.com(2600:9000:25f3:6200:7:49a5:5fd4:b121
 (2600:9000:25f3:6200:7:49a5:5fd4:b121)) 56 data bytes
```

请注意，运行 ping 命令失败并且流量将被阻止。

6. 连接到实例 D。由于没有可以运行 ping 命令的公有 IP 地址，因此请使用 EC2 Instance Connect 进行连接，然后从该实例对公有 IP 运行 ping 命令以检查出站流量：

```
aws ec2-instance-connect ssh --instance-id i-05f9e6a9cfac1dba0 --region us-
east-2 --connection-type eice
```

输出：

```
A newer release of "Amazon Linux" is available.  Version 2023.5.20240916:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Sep 27 18:20:58 2024 from 3.16.146.5
[ec2-user@ip-10-0-3-59 ~]$ ping www.amazon.com
PING www.amazon.com(2600:9000:25f3:b400:7:49a5:5fd4:b121
 (2600:9000:25f3:b400:7:49a5:5fd4:b121)) 56 data bytes
```

请注意，运行 ping 命令失败并且流量将被阻止。

清理

在本节中，您将删除为此高级示例创建的所有资源。清理资源很重要，这样可以避免在您账户中创建的资源产生过多的额外费用。

删除 CloudFormation 资源

完成本节以删除您使用 AWS CloudFormation 模板创建的资源。

AWS Management Console

1. 从 AWS CloudFormation 打开 https://console.aws.amazon.com/cloudformation/ 控制台。
2. 选择 VPC BPA 堆栈。
3. 选择删除。
4. 开始删除堆栈后，请查看事件选项卡，以查看进度并确保堆栈已删除。您可能必须强制删除堆栈才能将其完全删除。

AWS CLI

1. 删除 CloudFormation 堆栈。您可能必须强制删除堆栈才能将其完全删除。

```
aws cloudformation delete-stack --stack-name VPC-BPA-stack --region us-east-2
```

2. 查看进度并确保堆栈已删除。

```
aws cloudformation describe-stack-events --stack-name VPC-BPA-stack --region us-east-2
```

使用 AWS CloudTrail 追踪排除项删除

完成本节以使用 AWS CloudTrail 追踪排除项删除。当您删除排除项时，会显示 CloudTrail 条目。

AWS Management Console

通过在位于 https://console.aws.amazon.com/cloudtrailv2/ 的 AWS CloudTrail 控制台中查找资源类型 > AWS::EC2::VPCBlockPublicAccessExclusion，您可以在 CloudTrail 事件历史记录中查看任何已删除的排除项。

AWS CLI

您可以使用 lookup-events 命令查看与删除排除项有关的事件：

```
aws cloudtrail lookup-events --lookup-attributes
 AttributeKey=ResourceType,AttributeValue=AWS::EC2::VPCBlockPublicAccessExclusion
```

高级示例已完成。

VPC 的安全最佳实践

以下最佳实践是一般指导原则，并不代表完整安全解决方案。这些最佳实践可能不适合环境或不满足环境要求，请将其视为有用的考虑因素而不是惯例。

- 向 VPC 添加子网以托管应用程序时，请在多个可用区中创建子网。可用区是 AWS 区域中一个或多个具有冗余电源、网络和连接的离散数据中心。使用可用区可为生产级应用程序提供高可用性、容错能力和可扩展性。
- 使用安全组来控制流向子网中的 EC2 实例的流量。有关更多信息，请参阅 安全组。
- 使用网络 ACL 在子网级别控制入站和出站流量。有关更多信息，请参阅 使用网络访问控制列表控制子网流量。
- 使用 AWS Identity and Access Management（IAM）身份联合验证、用户和角色，管理对 VPC 中 AWS 资源的访问。有关更多信息，请参阅 适用于 Amazon VPC 的 Identity and Access Management。
- 使用 VPC 流日志监控传入和传出 VPC、子网或网络接口的 IP 流量。有关更多信息，请参阅 VPC 流日志。
- 使用网络访问分析器识别对我们 VPC 中资源的计划外网络访问。有关更多信息，请参阅 网络访问分析器用户指南。
- 使用 AWS Network Firewall 通过筛选入站和出站流量来监控和保护您的 VPC。有关更多信息，请参阅 AWS Network Firewall 指南。
- 使用 Amazon GuardDuty 检测您的账户、容器、工作负载以及 AWS 环境中的数据面临的潜在威胁。基础威胁检测包括监控与 Amazon EC2 实例关联的 VPC 流日志。有关更多信息，请参阅《Amazon GuardDuty 用户指南》中的 VPC 流日志。

有关 VPC 安全性的常见问题的答案，请参阅 Amazon VPC 常见问题解答中的安全性和筛选。

将 Amazon VPC 与其他 AWS 服务结合使用

Amazon Virtual Private Cloud（VPC）是一项基础 AWS 服务，可为您的云基础设施提供安全且可自定义的联网环境。除了创建并管理自有 VPC，您还可以利用 VPC 与其他 AWS 服务之间的集成来构建针对自身特定需求的全面解决方案。

您可以使用 AWS PrivateLink 将自己的 VPC 连接到各种 AWS 服务。这样可以实现自己的 VPC 与支持的 AWS 服务或本地应用程序之间的私有连接，从而将网络流量保留在 AWS 网络内，避免暴露于公共互联网。这对于维护严格的安全边界和合规性要求尤为有用。

要进一步加强 VPC 的安全，您可以使用 AWS Network Firewall。您可以借助此托管防火墙服务定义并强制实施网络级安全策略，筛选 VPC 内的南北向和东西向流量。通过将 Network Firewall 与自己的 VPC 配对，您可以增强防御策略，保护自己的云资源免受未经授权的访问或恶意活动的侵害。

此外，您可以使用 Route 53 Resolver DNS Firewall 筛选 VPC 内的 DNS 流量。您可借助此功能创建自定义 DNS 筛选规则来控制自己的 VPC 资源可以解析哪些域，从而提供额外的安全性与合规性强制执行层。

如果遇到 VPC 内部资源或连接到 VPC 的资源之间的可达性问题，则可使用 Reachability Analyzer。Reachability Analyzer 执行虚拟连接测试，提供详细的逐跳路径信息并识别任何阻碍组件。此故障排除工具有助于快速识别并解决网络连接问题。

通过将这些补充 AWS 服务与自己的 VPC 集成，您可以构建强大且安全的弹性云解决方案，可满足自身独特业务和架构要求。

内容

- 使用 AWS PrivateLink 将 VPC 连接到服务
- 使用 AWS Network Firewall 筛选网络流量
- 使用 Route 53 Resolver DNS Firewall 筛选 DNS 流量
- 使用 Reachability Analyzer 排查可达性问题

使用 AWS PrivateLink 将 VPC 连接到服务

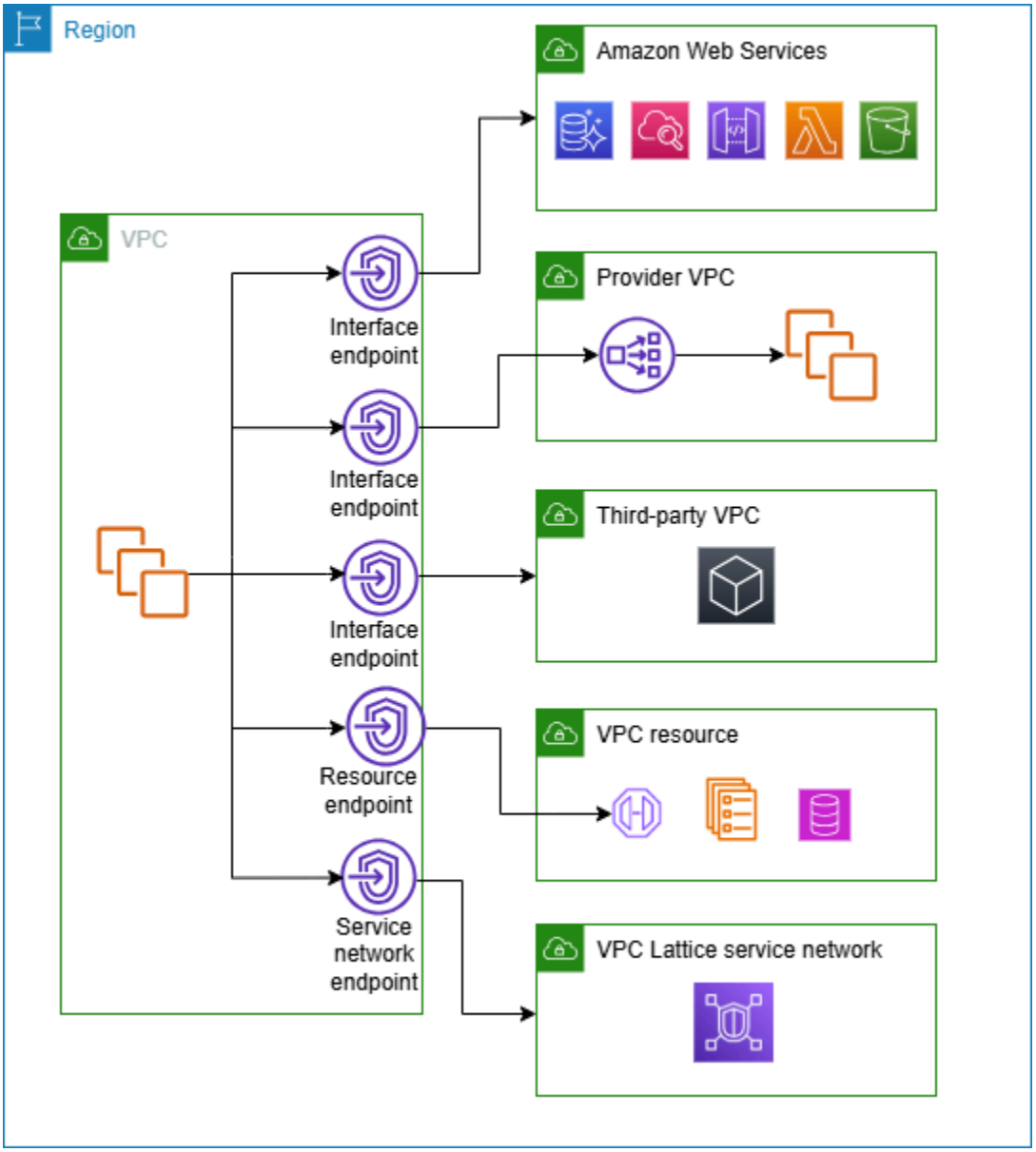
AWS PrivateLink 可在虚拟私有云 (VPC) 和支持的 AWS 服务、其他 AWS 账户托管的服务、支持的 AWS Marketplace 服务以及支持的资源之间建立私有连接。您无需使用互联网网关、NAT 设备、AWS Direct Connect 连接或 AWS Site-to-Site VPN 连接，就能与该服务或资源通信。

要使用 AWS PrivateLink，请在需要访问服务或资源的任何子网中创建 VPC 端点。这会在指定子网中创建弹性网络接口，作为发往服务或资源的流量的入口点。

您可以创建由 AWS PrivateLink 提供支持的自有 VPC 端点服务，让其他 AWS 客户能够访问您的服务。PrivateLink 支持创建私有 API 端点，方便组织安全地向其他 AWS 客户公开自己的服务。这使企业能够在内部功能上实现收益、促进协作生态系统，并控制其服务的访问和使用方式。

使用 AWS PrivateLink 的一项主要优势是能够建立安全的私有连接，无需互联网网关、NAT 设备或 VPN 连接等传统联网结构。此方法将数据流量限制在 AWS 网络内部，有助于简化网络架构、减少攻击面，继而提高整体安全性。

下图显示了 AWS PrivateLink 的常见用例。VPC 在私有子网中拥有多个 EC2 实例，这些实例可以通过五个 VPC 端点访问资源。有三个接口 VPC 端点、一个资源 VPC 端点和一个服务网络 VPC 端点。



有关更多信息，请参阅 AWS PrivateLink。

使用 AWS Network Firewall 筛选网络流量

您可以使用 AWS Network Firewall 在 VPC 外围筛选网络流量。Network Firewall 是一项有状态、托管的网络防火墙和入侵检测和防御服务。有关更多信息，请参阅 AWS Network Firewall 开发人员指南。

您可以使用以下 AWS 资源实施 Network Firewall。

Network Firewall 资源	描述
防火墙	防火墙将防火墙策略的网络流量筛选行为连接到要保护的 VPC。防火墙配置包括放置防火墙终端节点所在的可用区和子网的规范。它还定义了高级别的设置，例如防火墙日志记录配置和 AWS 防火墙资源上的标记。 有关更多信息，请参阅 AWS Network Firewall 中的防火墙。
防火墙策略	防火墙策略定义防火墙的监控和保护行为。行为的详细信息在添加到策略的规则组和某些策略默认设置中定义。要使用防火墙策略，请将其与一个或多个防火墙关联。 有关更多信息，请参阅 AWS Network Firewall 中的防火墙策略。
规则组	规则组是检查和处理网络流量的一组可重复使用的标准。作为策略配置的一部分，您可以将一个或多个规则组添加到防火墙策略中。您可以定义无状态规则组来检查隔离中的每个网络数据包。无状态规则组在行为和使用上类似于 Amazon VPC 网络访问控制列表 (ACL)。您还可以定义有状态的规则组，以便在数据包的流量上下文中检查数据包。有状态规则组在行为和使用方面类似于 Amazon VPC 安全组。 有关规则组的更多信息，请参阅 AWS Network Firewall 中的规则组。

您还可以使用 AWS Firewall Manager 集中配置和管理 AWS Organizations 中的跨账户和应用程序 Network Firewall 资源。您可以在 Firewall Manager 中使用单个账户管理多个账户的防火墙。有关更多信息，请参阅《AWS WAF、AWS Firewall Manager 和 AWS Shield Advanced 开发人员指南》中的 AWS Firewall Manager。

使用 Route 53 Resolver DNS Firewall 筛选 DNS 流量

使用 DNS Firewall，您可以在与您的 VPC 关联的规则组中定义域名过滤规则。您可以指定要允许或阻挡的域名列表，也可以自定义对阻挡的 DNS 查询的响应。有关详细信息，请参阅 Route 53 解析器 DNS 防火墙文档。

您可以使用以下 AWS 资源实施 DNS 防火墙。

DNS 防火墙资源	描述
DNS 防火墙规则组	DNS 防火墙规则组是用于过滤 DNS 查询的 DNS 防火墙规则的命名、可重复使用的集合。您可以使用过滤规则填充规则组，然后将规则组与 Amazon VPC 中的一个或多个 VPC 关联。当您关联规则组与 VPC 时，您可以为 VPC 启用 DNS 防火墙过滤。然后，当解析器收到与其关联的规则组的 VPC 的 DNS 查询时，解析器将该查询传递给 DNS 防火墙进行过滤。 规则组中的每个规则都指定一个域列表和要对域与列表中的域规范匹配的 DNS 查询执行的操作。您可以允许、阻挡匹配查询或发出警报。您还可以为阻挡的查询定义自定义响应。 有关详细信息，请参阅 Route 53 解析器 DNS 防火墙中的规则组和规则。
域列表	域列表是您在规则组内的 DNS 防火墙规则中使用的一组可重复使用的域规范。 有关详细信息，请参阅 Route 53 解析器 DNS 防火墙中的域列表。

您还可以使用 AWS Firewall Manager 集中配置和管理 AWS Organizations 中跨账户和企业的 DNS 防火墙资源。您可以在 Firewall Manager 中使用单个账户管理多个账户的防火墙。有关更多信息，请参阅《AWS WAF、AWS Firewall Manager 和 AWS Shield Advanced 开发人员指南》中的 AWS Firewall Manager。

使用 Reachability Analyzer 排查可达性问题

Reachability Analyzer 是一款静态配置分析工具。使用 Reachability Analyzer 可分析和调试 VPC 中两个资源之间的网络可达性。如果可以访问这些资源，则 Reachability Analyzer 会生成有关这些资源间虚拟路径的逐跳详细信息，否则会确定障碍组件。

您可以使用 Reachability Analyzer 分析以下资源之间的可达性：

- 实例
- Internet 网关
- 网络接口

- 中转网关
- 中转网关挂载
- VPC 端点服务
- VPC 端点
- VPC 对等连接
- VPN 网关

有关更多信息，请参阅 Reachability Analyzer 角色指南。

VPC 示例

Amazon Virtual Private Cloud（VPC）是 AWS 生态系统中的基本构建块，有助于您针对自身特定需求预置隔离的虚拟网络。您可以通过创建并管理自己的 VPC 来完全控制联网环境，包括定义 IP 地址范围、子网、路由表和连接选项的能力。

本节旨在介绍虚拟私有云（VPC）的三个配置示例，且每个示例都满足一组不同的需求：

- 面向测试环境的 VPC：此配置展示如何创建可用作开发或测试环境的 VPC。
- 面向 Web 服务器和数据库服务器的 VPC：此配置展示如何创建可用于生产环境中弹性架构的 VPC。
- 服务器位于私有子网和 NAT 中的 VPC：在这种更高级的配置中，所有 EC2 实例都在私有子网中预置，而 NAT 网关则有助于实现安全的出站互联网访问。在本示例中，您既要限制与资源的直接互联网连接，又要启用必要的出站通信。

我们提供这些 VPC 配置示例是希望说明在设计云联网环境时可用的灵活性和自定义选项。您选择的具体 VPC 设置应基于应用程序的架构、安全要求和整体业务目标。仔细规划 VPC 基础设施有助于您创建强大、可扩展且安全的虚拟网络，从而支持基于云的工作负载的增长和演变。

示例

- 示例：用于测试环境的 VPC
- 示例：用于 Web 和数据库服务器的 VPC
- 示例：在私有子网中部署服务器并且具有 NAT 中的 VPC

相关示例

- 要将您的 VPC 相互连接，请参阅《Amazon VPC 对等连接指南》中的 VPC 对等连接配置。
- 要将 VPC 连接到您自己的网络，请参阅《AWS Site-to-Site VPN 用户指南》中的 Site-to-Site VPN scenarios。
- 要将您的 VPC 相互连接并连接到您自己的网络，请参阅《Amazon VPC 中转网关》中的 Example transit gateway scenarios。

其他资源

- 了解故障恢复模式和权衡（AWS 架构博客）

- 规划网络拓扑结构（AWS Well-Architected Framework）
- Amazon Virtual Private Cloud 连接选项（AWS 白皮书）

示例：用于测试环境的 VPC

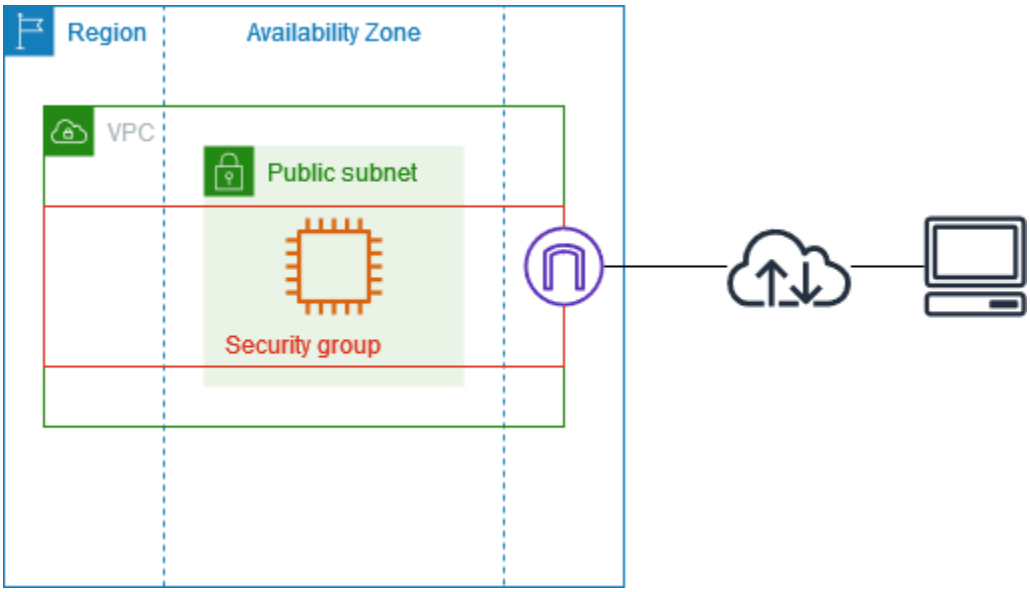
此示例将演示如何创建可用作开发或测试环境的 VPC。由于此 VPC 不旨在生产中使用，因此无需在多个可用区部署服务器。为了保持较低的成本和复杂性，您可以在单个可用区中部署服务器。

内容

- 概述
- 1. 创建 VPC
- 2. 部署您的应用程序
- 3. 测试配置
- 4. 清理

概述

下图概括了此例中包含的资源。此 VPC 在单个可用区中有一个公有子网，此外还有一个互联网网关。服务器是在公有子网中运行的一个 EC2 实例。该实例的安全组允许来自您自己计算机的 SSH 流量，以及您的开发或测试活动特别需要的任何其他流量。



路由

当您使用 Amazon VPC 控制台创建此 VPC 时，我们会为公有子网创建一个路由表，其中包含本地路由和指向互联网网关的路由。以下是一个路由表示例，其中同时包含了 IPv4 和 IPv6 路由。如果您创建仅 IPv4 子网而不是双堆栈子网，则您的路由表中将仅包含 IPv4 路由。

目标位置	目标
10.0.0.0/16	本地
2001:db8:1234:1a00::/56	本地
0.0.0.0/0	*igw-id*
::/0	*igw-id*

安全性

在此示例配置中，您必须为实例创建允许应用程序所需流量的安全组。例如，您可能需要添加一条规则，允许来自计算机的 SSH 流量或来自网络的 HTTP 流量。

以下是一个安全组入站规则示例，其中同时包含了 IPv4 和 IPv6 的规则。如果您创建仅 IPv4 子网而不是双堆栈子网，则仅需要 IPv4 的规则。

来源	协议	端口范围	描述
0.0.0.0/0	TCP	80	允许从所有 IPv4 地址进行入站 HTTP 访问
::/0	TCP	80	允许从所有 IPv6 地址进行入站 HTTP 访问
0.0.0.0/0	TCP	443	允许从所有 IPv4 地址进行入站 HTTPS 访问
::/0	TCP	443	允许从所有 IPv6 地址进行入站 HTTPS 访问

来源	协议	端口范围	描述
###### IPv4 ####	TCP	22	（可选）允许来自您网络中 IPv4 IP 地址的入站 SSH 访问
#### IPv6 ####	TCP	22	（可选）允许来自您网络中 IPv6 IP 地址的入站 SSH 访问
###### IPv4 ####	TCP	3389	（可选）允许来自您网络中 IPv4 IP 地址的入站 RDP 访问
#### IPv6 ####	TCP	3389	（可选）允许来自您网络中 IPv6 IP 地址的入站 RDP 访问

1. 创建 VPC

按照以下过程创建在一个可用区内具有一个公有子网的 VPC。此配置适合开发或测试环境。

创建 VPC

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在控制面板上，选择创建 VPC。
3. 对于要创建的资源，选择 VPC 等。
4. 配置 VPC
   a. 对于 Name tag auto-generation（名称标签自动生成），为 VPC 输入名称。
   b. 对于 IPv4 CIDR 块，您可以保留默认建议，也可以输入应用程序或网络所需的 CIDR 块。有关更多信息，请参阅 the section called “VPC CIDR 块”。
   c. （可选）如果应用程序使用 IPv6 地址进行通信，则选择 IPv6 CIDR 块、Amazon 提供的 IPv6 CIDR 块。
5. 配置子网
   a. 对于可用区数量，选择 1。您可以保留默认可用区，也可以展开自定义可用区并选择可用区。
   b. 对于 Number of public subnets（公有子网数量），选择 1。
   c. 对于 Number of private subnets（私有子网数量），选择 0。

d. 您可以保留公有子网的默认 CIDR 块，也可以展开自定义子网 CIDR 块并输入 CIDR 块。有关更多信息，请参阅 the section called “子网 CIDR 块”。

6. 对于 NAT 网关，请保留默认值（无）。
7. 对于 VPC endpoints（VPC 端点），选择 None（无）。S3 网关 VPC 端点仅用于从私有子网访问 Amazon S3。
8. 对于 DNS 选项，请同时选中这两个选项。这样可确保您的实例会收到与其公有 IP 地址对应的公有 DNS 主机名。
9. 选择创建 VPC。

2. 部署您的应用程序

您可以通过多种方式来部署 EC2 实例。例如：

- Amazon EC2 启动实例向导
- Amazon EC2 Auto Scaling
- AWS CloudFormation
- Amazon Elastic Container Service（Amazon ECS）

部署 EC2 实例后，您可以连接到该实例，安装应用程序所需的软件，然后创建映像以备将来之用。有关更多信息，请参阅《Amazon EC2 用户指南》中的创建 AMI。您还可以使用 EC2 Image Builder 来创建和管理您的亚马逊云机器镜像（AMI）。

3. 测试配置

完成应用程序部署后，您可以对其进行测试。如果无法连接到您的 EC2 实例，或者应用程序无法按照预期发送或接收流量，您可以使用 Reacability Analyzer 来帮助排查问题。例如，Reachability Analyzer 可以识别路由表或安全组配置问题。有关更多信息，请参阅 Reachability Analyzer 角色指南。

4. 清理

完成此配置后，您可以将其删除。您必须首先终止实例，然后才能删除 VPC。有关更多信息，请参阅 the section called “删除您的 VPC”。

示例：用于 Web 和数据库服务器的 VPC

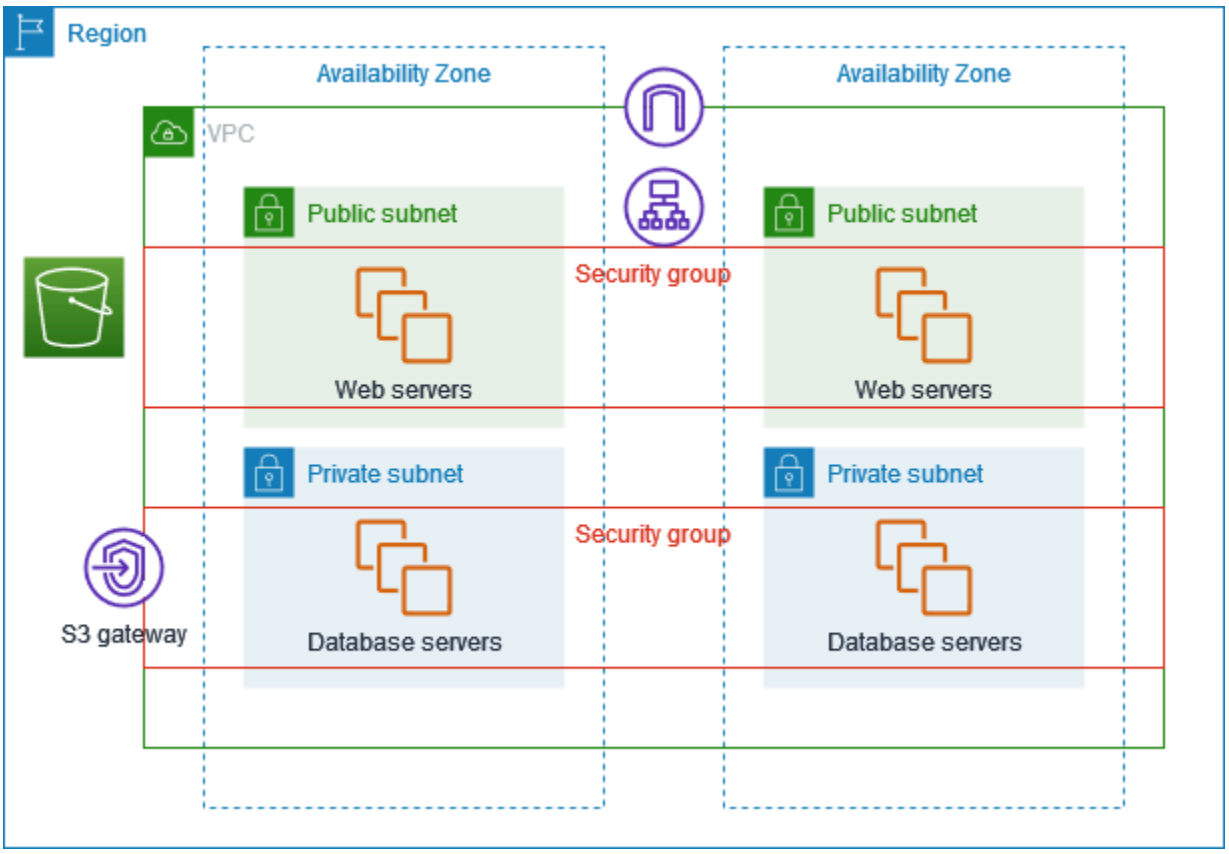
此示例将演示如何创建可用于生产环境的两层架构 VPC。为了提高故障恢复能力，您需要在两个可用区中部署服务器。

内容

- 概述
- 1. 创建 VPC
- 2. 部署您的应用程序
- 3. 测试配置
- 4. 清理

概述

下图概括了此例中包含的资源。此 VPC 在两个可用区中拥有公有和私有子网。Web 服务器在公有子网中运行，并通过负载均衡器接收来自客户端的流量。Web 服务器的安全组允许来自负载均衡器的流量。数据库服务器在私有子网中运行，并接收来自 Web 服务器的流量。数据库服务器的安全组允许来自 Web 服务器的流量。数据库服务器可以使用网关 VPC 端点连接到 Amazon S3。



路由

当您使用 Amazon VPC 控制台创建此 VPC 时，我们会为公有子网创建一个路由表，其中包含本地路由和指向互联网网关的路由，并为每个私有子网创建一个路由表，其中包含本地路由和指向网关 VPC 端点的路由。

以下是公有子网的路由表示例，其中同时包含了 IPv4 和 IPv6 路由。如果您创建仅 IPv4 子网而不是双堆栈子网，则您的路由表中将仅包含 IPv4 路由。

目标位置	目标
10.0.0.0/16	本地
2001:db8:1234:1a00::/56	本地
0.0.0.0/0	*igw-id*
::/0	*igw-id*

以下是私有子网的路由表示例，其中同时包含了 IPv4 和 IPv6 本地路由。如果您创建仅 IPv4 子网，则您的路由表中将仅包含 IPv4 路由。最后一条路由会将指向 Amazon S3 的流量发送到网关 VPC 端点。

目标位置	目标
10.0.0.0/16	本地
2001:db8:1234:1a00::/56	本地
s3-prefix-list-id	*s3-gateway-id*

安全性

对于此示例配置，您必须为负载均衡器创建一个安全组，为 Web 服务器创建一个安全组，并且为数据库服务器创建一个安全组。

负载均衡器

应用程序负载均衡器或网络负载均衡器的安全组必须允许来自负载均衡器侦听器端口上的客户端的入站流量。要接受来自互联网任何位置的流量，请指定 0.0.0.0/0 作为来源。负载均衡器安全组还必须在实例侦听器端口和运行状况检查端口上允许从负载均衡器到目标实例的出站流量。

Web 服务器

以下安全组规则允许 Web 服务器接收来自负载均衡器的 HTTP 和 HTTPS 流量。您还可以选择允许 Web 服务器接收来自您网络的 SSH 或 RDP 流量。Web 服务器可将 SQL 或 MySQL 流量发送到数据库服务器。

来源	协议	端口范围	描述
######### ID	TCP	80	允许来自负载均衡器的入站 HTPP 访问
######### ID	TCP	443	允许来自负载均衡器的入站 HTTPS 访问
###### IPv4 ####	TCP	22	（可选）允许来自您网络中 IPv4 IP 地址的入站 SSH 访问
#### IPv6 ####	TCP	22	（可选）允许来自您网络中 IPv6 IP 地址的入站 SSH 访问
###### IPv4 ####	TCP	3389	（可选）允许来自您网络中 IPv4 IP 地址的入站 RDP 访问
#### IPv6 ####	TCP	3389	（可选）允许来自您网络中 IPv6 IP 地址的入站 RDP 访问

目标位置	协议	端口范围	描述
## Microsoft SQL Server ####### ID	TCP	1433	允许对数据库服务器进行出站 Microsoft SQL Server 访问
## MySQL ####### ID	TCP	3306	允许对数据库服务器进行出站 MySQL 访问

数据库服务器

以下安全组规则允许数据库服务器接收来自 Web 服务器的读写请求。

来源	协议	端口范围	注释
Web ####### ID	TCP	1433	允许来自 Web 服务器的入站 Microsoft SQL Server 访问
Web ####### ID	TCP	3306	允许来自 Web 服务器的入站 MySQL Server 访问

目标位置	协议	端口范围	注释
0.0.0.0/0	TCP	80	允许通过 IPv4 到互联网的出站 HTTP 访问
0.0.0.0/0	TCP	443	允许通过 IPv4 到互联网的出站 HTTPS 访问

有关 Amazon RDS 数据库实例的安全组的更多信息，请参阅 Amazon RDS 用户指南 中的使用安全组控制访问。

1. 创建 VPC

按照以下过程创建在两个可用区内具有一个公有子网和一个私有子网的 VPC。

创建 VPC

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在控制面板上，选择创建 VPC。
3. 对于要创建的资源，选择 VPC 等。
4. 配置 VPC：
    a. 保持选中名称标签自动生成以为 VPC 资源创建名称标签，或者清除此选项以为 VPC 资源提供您自己的名称标签。

b. 对于 IPv4 CIDR 块，您可以保留默认建议，也可以输入应用程序或网络所需的 CIDR 块。有关更多信息，请参阅 the section called “VPC CIDR 块”。

c. （可选）如果应用程序使用 IPv6 地址进行通信，则选择 IPv6 CIDR 块、Amazon 提供的 IPv6 CIDR 块。

d. 选择租赁选项。此选项定义您启动到此 VPC 中的 EC2 实例是在与其他 AWS 账户 共享的硬件上运行，还是在专供您使用的硬件上运行。如果您选择将 VPC 的租赁设为 `Default`，则在此 VPC 中启动的 EC2 实例将使用您在启动实例时指定的租赁属性。有关更多信息，请参阅《Amazon EC2 用户指南》中的使用定义的参数启动实例。如果您选择 VPC 的租赁为 `Dedicated`，则这些实例将始终在专供您使用的硬件上作为专用实例运行。

5. 配置子网：

a. 对于可用区数量，选择 2，这样您可以在两个可用区中启动实例，以提高故障恢复能力。

b. 对于Number of public subnets（公有子网数量），选择 2。

c. 对于 Number of private subnets（私有子网数量），选择 2。

d. 您可以保留子网的默认 CIDR 块，也可以展开自定义子网 CIDR 块并输入 CIDR 块。有关更多信息，请参阅 the section called “子网 CIDR 块”。

6. 对于 NAT 网关，请保留默认值（无）。

7. 对于 VPC 端点，请保留默认值（S3 网关）。除非您要访问 S3 存储桶，此设置不会产生任何影响，不过启用此 VPC 端点是免费的。

8. 对于 DNS 选项，请同时选中这两个选项。这样可确保 Web 服务器会收到与其公有 IP 地址对应的公有 DNS 主机名。

9. 选择创建 VPC。

2. 部署您的应用程序

理想情况下，您已经在开发或测试环境中测试了 Web 服务器和数据库服务器，并创建了用于在生产环境中部署应用程序的脚本或映像。

您可以使用 EC2 实例来部署 Web 服务器。您可以通过多种方式来部署 EC2 实例。例如：

- Amazon EC2 启动实例向导
- AWS CloudFormation
- Amazon Elastic Container Service（Amazon ECS）

为了提高可用性，您可以使用 Amazon EC2 Auto Scaling 在多个可用区中部署服务器，并保持应用程序所需的最低服务器容量。

您可以使用 Elastic Load Balancing 在服务器之间均衡地分配流量。将负载均衡器附加到自动扩缩组。

您可以使用 EC2 实例来部署数据库服务器，也可以使用我们的任何一种专用数据库。有关更多信息，请参阅 AWS 上的数据库：如何选择。

3. 测试配置

完成应用程序部署后，您可以对其进行测试。如果应用程序无法按照预期发送或接收流量，您可以使用 Reacability Analyzer 来帮助排查问题。例如，Reachability Analyzer 可以识别路由表或安全组配置问题。有关更多信息，请参阅 Reachability Analyzer 角色指南。

4. 清理

完成此配置后，您可以将其删除。您必须首先终止实例并删除负载均衡器，然后才能删除 VPC。有关更多信息，请参阅 the section called “删除您的 VPC”。

示例：在私有子网中部署服务器并且具有 NAT 中的 VPC

此示例将演示如何创建可用于生产环境的服务器的 VPC。为了提高故障恢复能力，您需要使用一个自动扩缩组和一个应用程序负载均衡器，在两个可用区中部署服务器。为了提高安全性，您需要在私有子网中部署服务器。服务器通过负载均衡器接收请求。服务器可以使用 NAT 网关连接到互联网。为了提高故障恢复能力，您需要在这两个可用区中部署 NAT 网关。

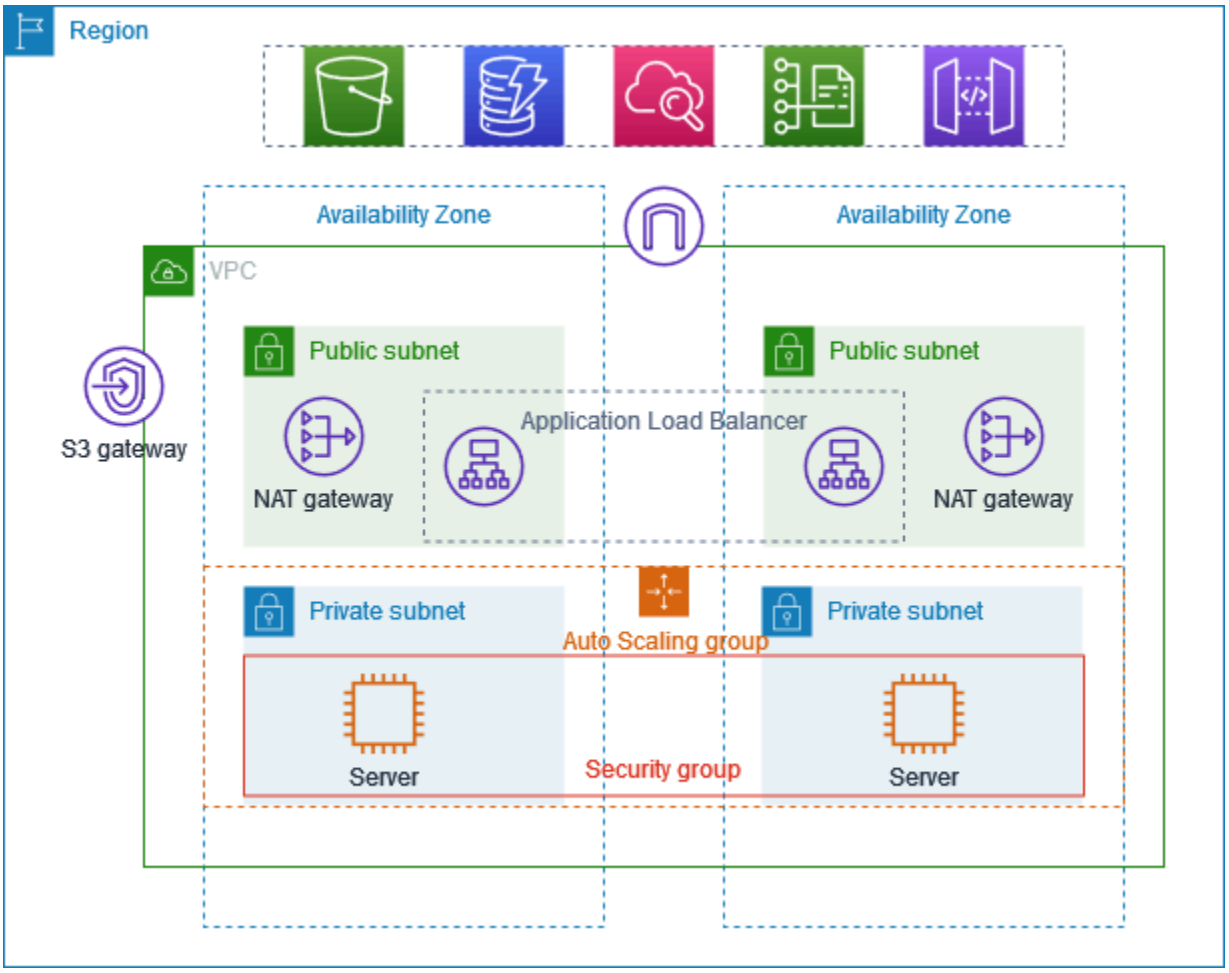
内容

- 概述
- 1. 创建 VPC
- 2. 部署您的应用程序
- 3. 测试配置
- 4. 清理

概述

下图概括了此例中包含的资源。此 VPC 在两个可用区中拥有公有和私有子网。每个公有子网都包含一个 NAT 网关和一个负载均衡器节点。服务器在私有子网中运行，使用自动扩缩组启动和终止，并接收

来自负载均衡器的流量。服务器可以使用 NAT 网关连接到互联网。服务器可以使用网关 VPC 端点连接到 Amazon S3。



路由

当您使用 Amazon VPC 控制台创建此 VPC 时，我们会为公有子网创建一个路由表，其中包含本地路由和指向互联网网关的路由。我们还将为私有子网创建一个路由表，其中包含本地路由，以及指向 NAT 网关、仅限出口的互联网网关和网关 VPC 端点的路由。

以下是公有子网的路由表示例，其中同时包含了 IPv4 和 IPv6 路由。如果您创建仅 IPv4 子网而不是双堆栈子网，则您的路由表中将仅包含 IPv4 路由。

目标位置	目标
10.0.0.0/16	本地
2001:db8:1234:1a00::/56	本地

目标位置	目标
0.0.0.0/0	*igw-id*
::/0	*igw-id*

以下是其中一个私有子网的路由表示例，其中同时包含了 IPv4 和 IPv6 路由。如果您创建仅 IPv4 子网，则您的路由表中将仅包含 IPv4 路由。最后一条路由会将指向 Amazon S3 的流量发送到网关 VPC 端点。

目标位置	目标
10.0.0.0/16	本地
2001:db8:1234:1a00::/56	本地
0.0.0.0/0	*nat-gateway-id*
::/0	*eigw-id*
s3-prefix-list-id	*s3-gateway-id*

安全性

以下是您可能为服务器所关联安全组创建的规则示例。安全组必须允许从负载均衡器通过侦听器端口和协议的流量。此外，安全组还必须允许运行状况检查流量。

来源	协议	端口范围	注释
######### ID	*#####*	*#####*	允许从负载均衡器通过侦听器端口的入站流量
######### ID	*########*	*########*	允许来自负载均衡器的入站运行状况检查流量

1. 创建 VPC

按照以下过程创建在两个可用区内具有一个公有子网和一个私有子网的 VPC，并在每个可用区中有一个 NAT 网关。

创建 VPC

1. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
2. 在控制面板上，选择创建 VPC。
3. 对于要创建的资源，选择 VPC 等。
4. 配置 VPC
   a. 对于 Name tag auto-generation（名称标签自动生成），为 VPC 输入名称。
   b. 对于 IPv4 CIDR 块，您可以保留默认建议，也可以输入应用程序或网络所需的 CIDR 块。
   c. 如果应用程序使用 IPv6 地址进行通信，则选择 IPv6 CIDR 块、Amazon 提供的 IPv6 CIDR 块。
5. 配置子网
   a. 对于可用区数量，选择 2，这样您可以在多个可用区中启动实例，以提高故障恢复能力。
   b. 对于Number of public subnets（公有子网数量），选择 2。
   c. 对于 Number of private subnets（私有子网数量），选择 2。
   d. 您可以保留公有子网的默认 CIDR 块，也可以展开自定义子网 CIDR 块并输入 CIDR 块。有关更多信息，请参阅 the section called “子网 CIDR 块”。
6. 对于 NAT 网关，选择每个可用区 1 个以提高故障恢复能力。
7. 如果应用程序使用 IPv6 地址进行通信，对于仅限出口的互联网网关，请选择是。
8. 对于 VPC 端点，如果实例必须访问 S3 存储桶，请保留默认设置 S3 网关。否则，您私有子网中的实例将无法访问 Amazon S3。此选项不会产生任何费用，因此如果您未来可能使用 S3 存储桶，则可以保留默认值。如果您选择无，则以后可以随时添加网关 VPC 端点。
9. 对于 DNS 选项，请清除启用 DNS 主机名。
10. 选择创建 VPC。

2. 部署您的应用程序

理想情况下，您已经在开发或测试环境中完成了服务器测试，并创建了用于在生产环境中部署应用程序的脚本或映像。

您可以使用 Amazon EC2 Auto Scaling 在多个可用区部署服务器，并保持应用程序所需的最低服务器容量。

使用自动扩缩组启动实例

1. 创建启动模板以指定使用 Amazon EC2 Auto Scaling 启动 EC2 实例所需的配置信息。有关更多信息，请参阅《Amazon EC2 Auto Scaling 用户指南》中的 为自动扩缩组创建启动模板。
2. 创建一个自动扩缩组，这是具有最小、最大和所需大小的 EC2 实例的集合。有关更多信息，请参阅《Amazon EC2 Auto Scaling 用户指南》中的 使用启动模板创建自动扩缩组。
3. 创建一个负载均衡器，以在自动扩缩组中的实例之间均衡地分配流量，并将该负载均衡器附加到您的自动扩缩组。有关更多信息，请参阅 Elastic Load Balancing 用户指南 和《Amazon EC2 Auto Scaling 用户指南》中的 使用 Elastic Load Balancing。

3. 测试配置

完成应用程序部署后，您可以对其进行测试。如果应用程序无法按照预期发送或接收流量，您可以使用 Reacability Analyzer 来帮助排查问题。例如，Reachability Analyzer 可以识别路由表或安全组配置问题。有关更多信息，请参阅 Reachability Analyzer 角色指南。

4. 清理

完成此配置后，您可以将其删除。您必须首先删除自动扩缩组，终止实例，删除 NAT 网关并删除负载均衡器，然后才能删除 VPC。有关更多信息，请参阅 the section called “删除您的 VPC”。

VPC 教程

Amazon 虚拟私有云（VPC）是 AWS 中网络基础设施的基石。尽管 AWS 管理控制台提供了用户友好的界面，但 AWS 命令行界面（CLI）在创建和管理 VPC 资源方面提供了更高的灵活性和自动化能力。

本指南将介绍两种关键的 VPC 部署场景：

- 包含公有子网的基本 VPC 设置，适用于简单的 Web 应用程序
- 使用 NAT 网关、包含私有子网和公有子网的高级 VPC 配置，适用于多层应用程序

借助这些教程，您将了解如何：

- 创建具有不同子网配置的 VPC
- 设置互联网和 NAT 网关
- 配置路由表和安全组
- 使用 AWS CLI 命令管理网络基础设施
- 实施 AWS 联网最佳实践

让我们开始使用命令行构建 AWS 网络基础设施。

教程

- 开始通过 AWS CLI 使用 Amazon VPC
- 使用 AWS CLI 创建具有私有子网和 NAT 网关的 VPC

开始通过 AWS CLI 使用 Amazon VPC

本教程将指导您使用 AWS 命令行界面（AWS CLI）创建虚拟私有云（VPC）。您将了解如何设置包含公有子网和私有子网的 VPC、配置互联网连接，并部署 EC2 实例以演示常见的 Web 应用程序架构。

先决条件

在开始本教程之前，请确保您具有以下各项：

1. AWS CLI。如需安装，请遵循 AWS CLI 安装指南。
2. 已使用适当的凭证配置 AWS CLI。如果尚未设置凭证，请运行 `aws configure`。

3. 有关网络概念的基础知识。
4. 用于在 AWS 账户中创建和管理 VPC 资源的 适用于 Amazon VPC 的 Identity and Access Management。

成本考虑因素

本教程创建的 AWS 资源可能会让您的账户产生费用。费用主要来自 NAT 网关（每小时 0.045 美元，外加数据处理费用）和 EC2 实例（t2.micro，每个实例每小时约 0.0116 美元）。如果您在一小时内完成本教程并清理所有资源，则总费用约为 0.07 美元。要在开发环境中优化成本，可以考虑使用 NAT 实例代替 NAT 网关，这样可以显著降低成本。

在继续操作之前，让我们确认 AWS CLI 是否已正确配置。

```
aws configure list
```

您应该会看到 AWS 访问密钥、私有密钥和默认区域。此外，请确认您是否具有创建 VPC 资源所需的权限。

```
aws sts get-caller-identity
```

此命令会显示 AWS 账户 ID、用户 ID 和 ARN，确认您的凭证有效。

创建 VPC

虚拟私有云（VPC）是专用于您 AWS 账户的虚拟网络。在本部分中，您将创建 CIDR 块为 10.0.0.0/16 的 VPC，该 CIDR 块最多可提供 65536 个 IP 地址。

创建 VPC

以下命令会创建新的 VPC，并为其分配名称标签。

```
aws ec2 create-vpc --cidr-block 10.0.0.0/16 --tag-specifications
 'ResourceType=vpc,Tags=[{Key=Name,Value=MyVPC}]'
```

记下输出中的 VPC ID，以便在后续命令中使用。在本教程中，我们将使用“vpc-0123456789abcdef0”作为示例 VPC ID。在所有命令中，请将其替换为实际的 VPC ID。

启用 DNS 支持和主机名

默认情况下，新 VPC 中会禁用 DNS 解析和 DNS 主机名。启用这些功能让 VPC 中的实例可以解析域名。

```
aws ec2 modify-vpc-attribute --vpc-id vpc-0123456789abcdef0 --enable-dns-support
aws ec2 modify-vpc-attribute --vpc-id vpc-0123456789abcdef0 --enable-dns-hostnames
```

如果成功，这些命令不会产生输出。VPC 现已启用 DNS 支持和主机名解析。

创建子网

子网是 VPC 的 IP 地址范围分段，您可以在其中放置隔离的资源组。在本部分中，您将在两个可用区中创建公有子网和私有子网以实现高可用性。

获取可用的可用区

首先，检索所在区域中可用的可用区。

```
aws ec2 describe-availability-zones
```

在本教程中，我们将使用前两个可用区。记下它们在输出中的名称（例如“us-east-1a”和“us-east-1b”）。

创建公有子网

公有子网用于需要从互联网访问的资源，例如 Web 服务器。

```
aws ec2 create-subnet \
  --vpc-id vpc-0123456789abcdef0 \
  --cidr-block 10.0.0.0/24 \
  --availability-zone us-east-1a \
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=Public-Subnet-AZ1}]'
```

记下输出中的子网 ID。在本教程中，我们将以“subnet-0123456789abcdef0”作为第一个公有子网的示例。

```
aws ec2 create-subnet \
  --vpc-id vpc-0123456789abcdef0 \
  --cidr-block 10.0.1.0/24 \
  --availability-zone us-east-1b \
```

```
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=Public-Subnet-AZ2}]'
```

记下输出中的子网 ID。在本教程中，我们将以“subnet-0123456789abcdef1”作为第二个公有子网的示例。

创建私有子网

私有子网用于不应直接从互联网访问的资源，例如数据库。

```
aws ec2 create-subnet \
  --vpc-id vpc-0123456789abcdef0 \
  --cidr-block 10.0.2.0/24 \
  --availability-zone us-east-1a \
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=Private-Subnet-AZ1}]'
```

记下输出中的子网 ID。在本教程中，我们将以“subnet-0123456789abcdef2”作为第一个私有子网的示例。

```
aws ec2 create-subnet \
  --vpc-id vpc-0123456789abcdef0 \
  --cidr-block 10.0.3.0/24 \
  --availability-zone us-east-1b \
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=Private-Subnet-AZ2}]'
```

记下输出中的子网 ID。在本教程中，我们将以“subnet-0123456789abcdef3”作为第二个私有子网的示例。

您现在拥有四个子网：两个公有子网和两个私有子网，它们分布在两个可用区中。

提示：规划 CIDR 块时，请确保其不会与现有网络重叠。对于生产环境，请分配足够的 IP 地址以满足未来的增长需求，同时保持合理的子网大小以确保安全且便于管理。

配置互联网连接

要允许 VPC 中的资源与互联网通信，您需要创建并附加互联网网关。在本部分中，您将为 VPC 设置互联网连接。

创建 Internet 网关

通过互联网网关，可以实现 VPC 和互联网之间的通信。

```
aws ec2 create-internet-gateway \
  --tag-specifications 'ResourceType=internet-gateway,Tags=[{Key=Name,Value=MyIGW}]'
```

记下输出中的互联网网关 ID。在本教程中，我们将以“igw-0123456789abcdef0”作为示例。

将互联网网关附加到 VPC

创建互联网网关后，可将其附加到 VPC。

```
aws ec2 attach-internet-gateway --internet-gateway-id igw-0123456789abcdef0 --vpc-id
 vpc-0123456789abcdef0
```

创建和配置路由表

路由表包含决定了将网络流量定向到何处的规则（路由）。首先，为公有子网创建路由表。

```
aws ec2 create-route-table \
  --vpc-id vpc-0123456789abcdef0 \
  --tag-specifications 'ResourceType=route-table,Tags=[{Key=Name,Value=Public-RT}]'
```

记下输出中的路由表 ID。在本教程中，我们将以“rtb-0123456789abcdef0”作为公有路由表的示例。

在公有路由表中添加一条指向互联网网关的路由。

```
aws ec2 create-route --route-table-id rtb-0123456789abcdef0 --destination-cidr-block
 0.0.0.0/0 --gateway-id igw-0123456789abcdef0
```

将公有子网与公有路由表关联。

```
aws ec2 associate-route-table --route-table-id rtb-0123456789abcdef0 --subnet-id
 subnet-0123456789abcdef0
aws ec2 associate-route-table --route-table-id rtb-0123456789abcdef0 --subnet-id
 subnet-0123456789abcdef1
```

现在，为私有子网创建路由表。

```
aws ec2 create-route-table \
  --vpc-id vpc-0123456789abcdef0 \
  --tag-specifications 'ResourceType=route-table,Tags=[{Key=Name,Value=Private-RT}]'
```

记下输出中的路由表 ID。在本教程中，我们将以“rtb-0123456789abcdef1”作为私有路由表的示例。

将私有子网与私有路由表关联。

```
aws ec2 associate-route-table --route-table-id rtb-0123456789abcdef1 --subnet-id
 subnet-0123456789abcdef2
aws ec2 associate-route-table --route-table-id rtb-0123456789abcdef1 --subnet-id
 subnet-0123456789abcdef3
```

创建 NAT 网关

NAT 网关允许私有子网中的实例启动流向互联网的出站流量，同时阻止来自互联网的入站流量。对于需要下载更新或访问外部服务的实例而言，这至关重要。

分配弹性 IP

首先，为 NAT 网关分配弹性 IP 地址。

```
aws ec2 allocate-address --domain vpc
```

记下输出中的分配 ID。在本教程中，我们将以“eipalloc-0123456789abcdef0”作为示例。

创建 NAT 网关

使用分配的弹性 IP 在其中一个公有子网中创建 NAT 网关。

```
aws ec2 create-nat-gateway \
  --subnet-id subnet-0123456789abcdef0 \
  --allocation-id eipalloc-0123456789abcdef0 \
  --tag-specifications 'ResourceType=natgateway,Tags=[{Key=Name,Value=MyNATGateway}]'
```

记下输出中的 NAT 网关 ID。在本教程中，我们将以“nat-0123456789abcdef0”作为示例。

等待 NAT 网关变为可用后，再继续操作。

```
aws ec2 wait nat-gateway-available --nat-gateway-ids nat-0123456789abcdef0
```

向 NAT 网关添加路由

在私有路由表中添加一条指向 NAT 网关的路由，以允许私有子网中的实例访问互联网。

```
aws ec2 create-route --route-table-id rtb-0123456789abcdef1 --destination-cidr-block
 0.0.0.0/0 --nat-gateway-id nat-0123456789abcdef0
```

注意：对于生产环境，请考虑在每个拥有私有子网的可用区中创建一个 NAT 网关，以消除单点故障。

配置子网设置

配置公有子网，以便为在其中启动的实例自动分配公有 IP 地址。

```
aws ec2 modify-subnet-attribute --subnet-id subnet-0123456789abcdef0 --map-public-ip-
on-launch
aws ec2 modify-subnet-attribute --subnet-id subnet-0123456789abcdef1 --map-public-ip-
on-launch
```

这可确保在公有子网中启动的实例在默认情况下会收到公有 IP 地址，从而可从互联网访问这些实例。

创建安全组

安全组充当实例的虚拟防火墙以控制入站和出站流量。在本部分中，您将为 Web 服务器和数据库服务器创建安全组。

为 Web 服务器创建安全组

```
aws ec2 create-security-group \
  --group-name WebServerSG \
  --description "Security group for web servers" \
  --vpc-id vpc-0123456789abcdef0
```

记下输出中的安全组 ID。在本教程中，我们将以“sg-0123456789abcdef0”作为 Web 服务器安全组的示例。

允许 HTTP 和 HTTPS 流量访问 Web 服务器。

```
aws ec2 authorize-security-group-ingress --group-id sg-0123456789abcdef0 --protocol tcp
 --port 80 --cidr 0.0.0.0/0
aws ec2 authorize-security-group-ingress --group-id sg-0123456789abcdef0 --protocol tcp
 --port 443 --cidr 0.0.0.0/0
```

注意：对于生产环境，请将入站流量限制为特定 IP 范围，而不是允许来自 0.0.0.0/0（任何 IP 地址）的流量。

为数据库服务器创建安全组

```
aws ec2 create-security-group \
  --group-name DBServerSG \
  --description "Security group for database servers" \
  --vpc-id vpc-0123456789abcdef0
```

记下输出中的安全组 ID。在本教程中，我们将以“sg-0123456789abcdef1”作为数据库服务器安全组的示例。

仅允许来自 Web 服务器的 MySQL/Aurora 流量。

```
aws ec2 authorize-security-group-ingress --group-id sg-0123456789abcdef1 --protocol tcp
 --port 3306 --source-group sg-0123456789abcdef0
```

此配置遵循最低权限原则，确保只有 Web 服务器安全组中的实例才能通过端口 3306 连接到数据库服务器。

验证您的 VPC 配置

所有必要组件创建完成后，请验证 VPC 配置以确保各项均已正确设置。

检查 VPC

```
aws ec2 describe-vpcs --vpc-id vpc-0123456789abcdef0
```

检查子网

```
aws ec2 describe-subnets --filters "Name=vpc-id,Values=vpc-0123456789abcdef0"
```

检查路由表

```
aws ec2 describe-route-tables --filters "Name=vpc-id,Values=vpc-0123456789abcdef0"
```

检查互联网网关

```
aws ec2 describe-internet-gateways --filters "Name=attachment.vpc-
id,Values=vpc-0123456789abcdef0"
```

检查 NAT 网关

```
aws ec2 describe-nat-gateways --filter "Name=vpc-id,Values=vpc-0123456789abcdef0"
```

检查安全组

```
aws ec2 describe-security-groups --filters "Name=vpc-id,Values=vpc-0123456789abcdef0"
```

这些命令提供了有关 VPC 每个组件的详细信息，便于您验证各项是否配置正确。

部署 EC2 实例

现在，您已经创建 VPC 基础设施，可以部署 EC2 实例来演示架构如何工作。您将在公有子网中启动 Web 服务器，并在私有子网中启动数据库服务器。

创建用于访问 SSH 的密钥对

首先，创建密钥对以安全地连接到实例：

```
aws ec2 create-key-pair --key-name vpc-tutorial-key --query 'KeyMaterial' --output text
 > vpc-tutorial-key.pem
chmod 400 vpc-tutorial-key.pem
```

此命令会创建新的密钥对，并将私有密钥保存到具有受限权限的文件中。

查找最新的 Amazon Linux 2 AMI

查找用于实例的最新 Amazon Linux 2 AMI：

```
aws ec2 describe-images --owners amazon \
  --filters "Name=name,Values=amzn2-ami-hvm-*-x86_64-gp2" "Name=state,Values=available"
 \
  --query "sort_by(Images, &CreationDate)[-1].ImageId" --output text
```

记下输出中的 AMI ID。在本教程中，我们将以“ami-0123456789abcdef0”作为示例。

在公有子网中启动 Web 服务器

现在，在公有子网中启动 EC2 实例作为 Web 服务器：

```
aws ec2 run-instances \
  --image-id ami-0123456789abcdef0 \
  --count 1 \
  --instance-type t2.micro \
  --key-name vpc-tutorial-key \
  --security-group-ids sg-0123456789abcdef0 \
  --subnet-id subnet-0123456789abcdef0 \
  --associate-public-ip-address \
  --user-data '#!/bin/bash
               yum update -y
               yum install -y httpd
               systemctl start httpd
               systemctl enable httpd
               echo "<h1>Hello from $(hostname -f)</h1>" > /var/www/html/index.html' \
  --tag-specifications 'ResourceType=instance,Tags=[{Key=Name,Value=WebServer}]'
```

记下输出中的实例 ID。在本教程中，我们将使以“i-0123456789abcdef0”作为 Web 服务器实例的示例。

在私有子网中启动数据库服务器

接下来，在私有子网中启动 EC2 实例作为数据库服务器：

```
aws ec2 run-instances \
  --image-id ami-0123456789abcdef0 \
  --count 1 \
  --instance-type t2.micro \
  --key-name vpc-tutorial-key \
  --security-group-ids sg-0123456789abcdef1 \
  --subnet-id subnet-0123456789abcdef2 \
  --user-data '#!/bin/bash
               yum update -y
               yum install -y mariadb-server
               systemctl start mariadb
               systemctl enable mariadb' \
  --tag-specifications 'ResourceType=instance,Tags=[{Key=Name,Value=DBServer}]'
```

记下输出中的实例 ID。在本教程中，我们将以“i-0123456789abcdef1”作为数据库服务器实例的示例。

访问 Web 服务器

Web 服务器实例开始运行后，便可使用其公有 IP 地址对其进行访问：

```
aws ec2 describe-instances \
  --instance-ids i-0123456789abcdef0 \
  --query 'Reservations[0].Instances[0].PublicIpAddress' \
  --output text
```

此命令将输出 Web 服务器的公有 IP 地址。在本教程中，我们将以“203.0.113.10”作为示例。

现在，您可以在 Web 浏览器中打开此 URL：http://203.0.113.10

通过 SSH 连接到实例

要连接到 Web 服务器，请执行以下操作：

```
ssh -i vpc-tutorial-key.pem ec2-user@203.0.113.10
```

要连接到数据库服务器，需要先通过 SSH 连接到 Web 服务器，然后再连接到数据库服务器：

```
# Get the private IP of the database server
aws ec2 describe-instances \
  --instance-ids i-0123456789abcdef1 \
  --query 'Reservations[0].Instances[0].PrivateIpAddress' \
  --output text
```

此命令将输出数据库服务器的私有 IP 地址。在本教程中，我们将以“10.0.2.10”作为示例。

```
# First SSH to web server, then to database server
ssh -i vpc-tutorial-key.pem -A ec2-user@203.0.113.10
ssh ec2-user@10.0.2.10
```

这演示了您创建的网络架构：Web 服务器可公开访问，而数据库服务器只能从 VPC 内部访问。

故障排除

以下是创建 VPC 时可能遇到的一些常见问题及其解决方法：

CIDR 块重叠

如果您收到有关 CIDR 块重叠的错误，请确保 VPC 和子网的 CIDR 块不会与账户中的现有 VPC 或子网重叠。

权限错误

如果遇到权限错误，请验证 IAM 用户或角色是否具有创建和管理 VPC 资源所需的权限。您可能需要附加 AmazonVPCFullAccess 策略或创建具有所需权限的自定义策略。

资源限制

AWS 账户对您可以创建的 VPC、子网及其他资源的数量施加了默认限制。如果您达到这些限制，可通过 AWS Support Center 请求提高限制。

清理期间出现依赖项故障

清理资源时，如果尝试以错误顺序删除资源，可能会遇到依赖项错误。始终按照与创建顺序相反的顺序删除资源，从依赖程度最高的资源开始。

清理资源

完成 VPC 的使用后，可以清理资源以避免产生费用。按照与创建顺序相反的顺序删除资源，以正确处理依赖项。

终止 EC2 实例

```
aws ec2 terminate-instances --instance-ids i-0123456789abcdef0 i-0123456789abcdef1
aws ec2 wait instance-terminated --instance-ids i-0123456789abcdef0 i-0123456789abcdef1
```

删除密钥对

```
aws ec2 delete-key-pair --key-name vpc-tutorial-key
rm vpc-tutorial-key.pem
```

删除 NAT 网关

```
aws ec2 delete-nat-gateway --nat-gateway-id nat-0123456789abcdef0
aws ec2 wait nat-gateway-deleted --nat-gateway-ids nat-0123456789abcdef0
```

释放弹性 IP

```
aws ec2 release-address --allocation-id eipalloc-0123456789abcdef0
```

删除安全组

```
aws ec2 delete-security-group --group-id sg-0123456789abcdef1
aws ec2 delete-security-group --group-id sg-0123456789abcdef0
```

删除路由表

首先，查找路由表关联 ID：

```
aws ec2 describe-route-tables --route-table-id rtb-0123456789abcdef0
aws ec2 describe-route-tables --route-table-id rtb-0123456789abcdef1
```

然后取消路由表与子网的关联（将关联 ID 替换为输出中实际的 ID）：

```
aws ec2 disassociate-route-table --association-id rtbassoc-0123456789abcdef0
aws ec2 disassociate-route-table --association-id rtbassoc-0123456789abcdef1
aws ec2 disassociate-route-table --association-id rtbassoc-0123456789abcdef2
aws ec2 disassociate-route-table --association-id rtbassoc-0123456789abcdef3
```

然后删除路由表：

```
aws ec2 delete-route-table --route-table-id rtb-0123456789abcdef1
aws ec2 delete-route-table --route-table-id rtb-0123456789abcdef0
```

分离并删除互联网网关

```
aws ec2 detach-internet-gateway --internet-gateway-id igw-0123456789abcdef0 --vpc-id
 vpc-0123456789abcdef0
aws ec2 delete-internet-gateway --internet-gateway-id igw-0123456789abcdef0
```

删除子网

```
aws ec2 delete-subnet --subnet-id subnet-0123456789abcdef0
aws ec2 delete-subnet --subnet-id subnet-0123456789abcdef1
aws ec2 delete-subnet --subnet-id subnet-0123456789abcdef2
aws ec2 delete-subnet --subnet-id subnet-0123456789abcdef3
```

删除 VPC

```
aws ec2 delete-vpc --vpc-id vpc-0123456789abcdef0
```

投入生产

本教程旨在帮助您了解如何使用 AWS CLI 创建 VPC。对于生产环境，请考虑以下安全和架构最佳实践：

1. 安全组规则：将入站流量限制为特定 IP 范围内，而不是允许来自 0.0.0.0/0 的流量。
2. 高可用性：在拥有私有子网的每个可用区部署 NAT 网关，以消除单点故障。
3. 网络 ACL：实施网络 ACL 作为安全组之外的额外安全层。
4. VPC 流日志：启用 VPC 流日志来监控和分析网络流量模式。
5. 资源标记：实施全面的标记策略，以更好地管理资源。

有关构建生产就绪架构的更多信息，请参阅 AWS Well-Architected 框架和 AWS 安全最佳实践。

后续步骤

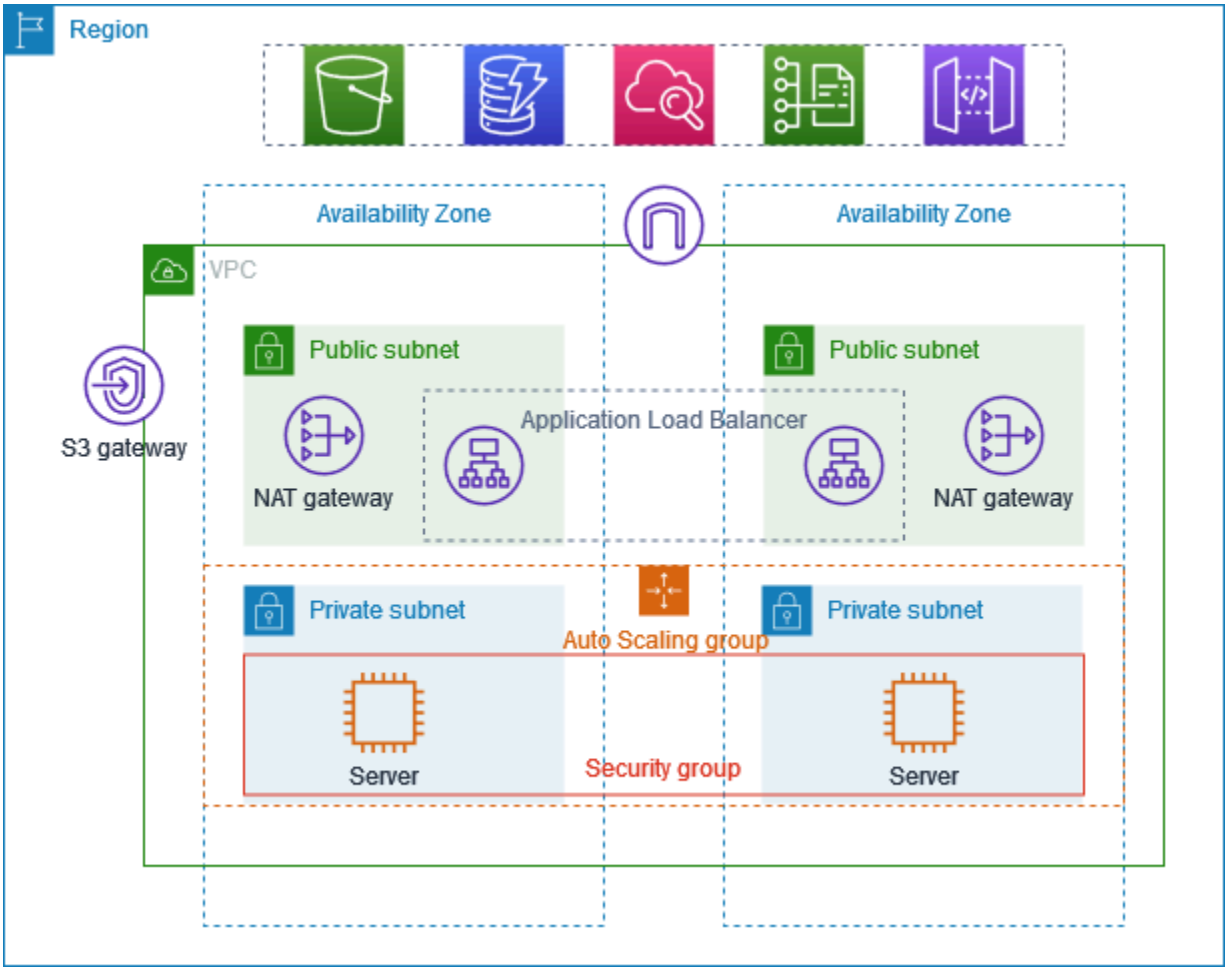
现在，您已创建包含公有子网和私有子网的 VPC，可以：

1. 在公有子网或私有子网中启动 EC2 实例。
2. 部署负载均衡器，以在多个实例之间分配流量。
3. 设置自动扩缩组，以实现高可用性和可扩展性。
4. 在私有子网中配置 RDS 数据库。
5. 实施 VPC 对等连接，以连接其他 VPC。
6. 设置 VPN 连接，以将 VPC 连接到本地网络。

使用 AWS CLI 创建具有私有子网和 NAT 网关的 VPC

本教程将演示如何使用 AWS CLI 创建可用于生产环境的服务器的 VPC。为了提高故障恢复能力，您需要使用一个自动扩缩组和一个应用程序负载均衡器在两个可用区中部署服务器。为了提高安全性，您需要在私有子网中部署服务器。服务器将通过负载均衡器接收请求，并且可以使用 NAT 网关连接到互联网。为了提高故障恢复能力，您需要在这两个可用区中分别部署 NAT 网关。

下图概括了本教程中包含的资源。此 VPC 在两个可用区中拥有公有和私有子网。每个公有子网都包含一个 NAT 网关和一个负载均衡器节点。服务器在私有子网中运行，使用自动扩缩组启动和终止，并接收来自负载均衡器的流量。服务器可以使用 NAT 网关连接到互联网。服务器可以使用网关 VPC 端点连接到 Amazon S3。



先决条件

开始本教程之前，您需要：

- 已安装并配置 AWS CLI，且具有创建 VPC 资源、EC2 实例、负载均衡器和自动扩缩组的权限。有关安装 AWS CLI 的信息，请参阅安装或更新 AWS CLI 的最新版本。
- 掌握 VPC 概念的基本知识，包括子网、路由表和互联网网关。
- 已安装 `jq` 命令行 JSON 处理器。这将用于解析 AWS CLI 命令的输出。有关安装 jq 的信息，请参阅 Download jq。
- 拥有要创建资源所需的足够服务配额，包括：
- 至少 2 个可用的弹性 IP 地址
- 至少 2 个 NAT 网关
- 至少 1 个 VPC
- 至少 4 个子网

- 至少 1 应用程序负载均衡器

估算费用：本教程中创建的资源将让您的 AWS 账户产生费用：NAT 网关：每小时约 0.045 美元，外加数据处理费用；弹性 IP 地址：与正在运行的实例关联时免费，未与 EC2 实例关联时每小时约 0.005 美元：费用因实例类型而异（本教程使用的是 t3.micro 实例）应用程序负载均衡器：每小时约 0.0225 美元，外加数据处理费用

创建 VPC 和子网

首先，您将创建 CIDR 块为 10.0.0.0/16 的 VPC，该 CIDR 块最多可提供 65536 个 IP 地址。

```
# Create a VPC with CIDR block 10.0.0.0/16
aws ec2 create-vpc --cidr-block 10.0.0.0/16 --tag-specifications
 'ResourceType=vpc,Tags=[{Key=Name,Value=ProductionVPC}]'
```

该命令返回的输出类似于下方内容：

```
{
    "Vpc": {
        "CidrBlock": "10.0.0.0/16",
        "DhcpOptionsId": "dopt-abcd1234",
        "State": "pending",
        "VpcId": "vpc-abcd1234",
        "OwnerId": "123456789012",
        "InstanceTenancy": "default",
        "Ipv6CidrBlockAssociationSet": [],
        "CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-abcd1234",
                "CidrBlock": "10.0.0.0/16",
                "CidrBlockState": {
                    "State": "associated"
                }
            }
        ],
        "IsDefault": false,
        "Tags": [
            {
                "Key": "Name",
                "Value": "ProductionVPC"
            }
        ]
```

```
    }
}
```

记下输出中的 VPC ID（例如 vpc-abcd1234）。您将在后续命令中使用此 ID。

接下来，您将在所在区域中确定两个可用区，用以创建弹性架构。

```
# Get available Availability Zones
aws ec2 describe-availability-zones --query 'AvailabilityZones[0:2].ZoneName' --output
 text
```

该命令返回的输出类似于下方内容：

```
us-east-1a us-east-1b
```

现在，创建四个子网：两个用于负载均衡器和 NAT 网关的公有子网，以及两个用于应用程序服务器的私有子网。将 vpc-abcd1234 替换为实际的 VPC ID，并将 us-east-1a 和 us-east-1b 替换为实际的可用区。

```
# Create public subnet in first AZ
aws ec2 create-subnet \
  --vpc-id vpc-abcd1234 \
  --cidr-block 10.0.0.0/24 \
  --availability-zone us-east-1a \
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=PublicSubnet1}]'

# Create private subnet in first AZ
aws ec2 create-subnet \
  --vpc-id vpc-abcd1234 \
  --cidr-block 10.0.1.0/24 \
  --availability-zone us-east-1a \
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=PrivateSubnet1}]'

# Create public subnet in second AZ
aws ec2 create-subnet \
  --vpc-id vpc-abcd1234 \
  --cidr-block 10.0.2.0/24 \
  --availability-zone us-east-1b \
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=PublicSubnet2}]'

# Create private subnet in second AZ
aws ec2 create-subnet \
```

```
  --vpc-id vpc-abcd1234 \
  --cidr-block 10.0.3.0/24 \
  --availability-zone us-east-1b \
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=PrivateSubnet2}]'
```

每个命令返回的输出均包含子网 ID。记下这些 ID，以便在后续命令中使用：

- 公有子网 1：subnet-abcd1234
- 私有子网 1：subnet-abcd5678
- 公有子网 2：subnet-efgh1234
- 私有子网 2：subnet-efgh5678

创建和配置互联网连接

在本部分中，您将创建互联网网关，以允许 VPC 与互联网之间进行通信，并将其附加到 VPC。

```
# Create an Internet Gateway
aws ec2 create-internet-gateway --tag-specifications 'ResourceType=internet-
gateway,Tags=[{Key=Name,Value=ProductionIGW}]'
```

该命令返回的输出包含互联网网关 ID。记下此 ID（例如 igw-abcd1234）。

将互联网网关附加到 VPC。将 igw-abcd1234 替换为实际的互联网网关 ID，并将 vpc-abcd1234 替换为实际的 VPC ID。

```
# Attach the Internet Gateway to the VPC
aws ec2 attach-internet-gateway --internet-gateway-id igw-abcd1234 --vpc-id vpc-
abcd1234
```

接下来，为公有子网与私有子网创建路由表。将 vpc-abcd1234 替换为实际的 VPC ID。

```
# Create a route table for public subnets
aws ec2 create-route-table --vpc-id vpc-abcd1234 --tag-specifications
 'ResourceType=route-table,Tags=[{Key=Name,Value=PublicRouteTable}]'

# Create route table for private subnet in first AZ
aws ec2 create-route-table --vpc-id vpc-abcd1234 --tag-specifications
 'ResourceType=route-table,Tags=[{Key=Name,Value=PrivateRouteTable1}]'

# Create route table for private subnet in second AZ
```

```
aws ec2 create-route-table --vpc-id vpc-abcd1234 --tag-specifications
 'ResourceType=route-table,Tags=[{Key=Name,Value=PrivateRouteTable2}]'
```

每个命令返回的输出均包含路由表 ID。记下这些 ID：

- 公有路由表：rtb-abcd1234
- 私有路由表 1：rtb-efgh1234
- 私有路由表 2：rtb-ijkl1234

在公有路由表中添加一条指向互联网网关的路由，以启用互联网访问。将 rtb-abcd1234 替换为实际的公有路由表 ID，并将 igw-abcd1234 替换为实际的互联网网关 ID。

```
# Add a route to the Internet Gateway
aws ec2 create-route --route-table-id rtb-abcd1234 --destination-cidr-block 0.0.0.0/0
 --gateway-id igw-abcd1234
```

将子网与其各自的路由表关联。将路由表 ID 和子网 ID 替换为实际的 ID。

```
# Associate public subnets with the public route table
aws ec2 associate-route-table --route-table-id rtb-abcd1234 --subnet-id subnet-abcd1234
aws ec2 associate-route-table --route-table-id rtb-abcd1234 --subnet-id subnet-efgh1234

# Associate private subnets with their respective route tables
aws ec2 associate-route-table --route-table-id rtb-efgh1234 --subnet-id subnet-abcd5678
aws ec2 associate-route-table --route-table-id rtb-ijkl1234 --subnet-id subnet-efgh5678
```

创建 NAT 网关

NAT 网关允许私有子网中的实例连接到互联网或其他 AWS 服务，但是阻止互联网启动与这些实例的连接。首先，为 NAT 网关分配弹性 IP 地址。

```
# Allocate Elastic IP for NAT Gateway in first AZ
aws ec2 allocate-address --domain vpc --tag-specifications 'ResourceType=elastic-
ip,Tags=[{Key=Name,Value=NAT1-EIP}]'

# Allocate Elastic IP for NAT Gateway in second AZ
aws ec2 allocate-address --domain vpc --tag-specifications 'ResourceType=elastic-
ip,Tags=[{Key=Name,Value=NAT2-EIP}]'
```

每个命令返回的输出均包含分配 ID。记下这些 ID：

- EIP 1 分配 ID：eipalloc-abcd1234
- EIP 2 分配 ID：eipalloc-efgh1234

在每个公有子网中创建 NAT 网关。将子网 ID 和分配 ID 替换为实际的 ID。

```
# Create NAT Gateway in public subnet of first AZ
aws ec2 create-nat-gateway \
  --subnet-id subnet-abcd1234 \
  --allocation-id eipalloc-abcd1234 \
  --tag-specifications 'ResourceType=natgateway,Tags=[{Key=Name,Value=NAT-Gateway1}]'

# Create NAT Gateway in public subnet of second AZ
aws ec2 create-nat-gateway \
  --subnet-id subnet-efgh1234 \
  --allocation-id eipalloc-efgh1234 \
  --tag-specifications 'ResourceType=natgateway,Tags=[{Key=Name,Value=NAT-Gateway2}]'
```

每个命令返回的输出均包含 NAT 网关 ID。记下这些 ID：

- NAT 网关 1：nat-abcd1234
- NAT 网关 2：nat-efgh1234

NAT 网关需要几分钟时间预置。等待 NAT 网关变为可用后，再继续操作。将 NAT 网关 ID 替换为实际的 ID。

```
# Wait for NAT Gateways to be available
aws ec2 wait nat-gateway-available --nat-gateway-ids nat-abcd1234
aws ec2 wait nat-gateway-available --nat-gateway-ids nat-efgh1234
```

在私有路由表中添加指向 NAT 网关的路由，以允许私有子网中的实例访问互联网。将路由表 ID 和 NAT 网关 ID 替换为实际的 ID。

```
# Add route to NAT Gateway 1 in private route table 1
aws ec2 create-route \
  --route-table-id rtb-efgh1234 \
  --destination-cidr-block 0.0.0.0/0 \
  --nat-gateway-id nat-abcd1234

# Add route to NAT Gateway 2 in private route table 2
aws ec2 create-route \
```

```
  --route-table-id rtb-ijkl1234 \
  --destination-cidr-block 0.0.0.0/0 \
  --nat-gateway-id nat-efgh1234
```

为 Amazon S3 创建 VPC 端点

Amazon S3 的 VPC 端点允许私有子网中的实例无需通过 NAT 网关即可访问 S3，这可以降低数据传输成本并提供更好的网络性能。将 vpc-abcd1234 替换为实际的 VPC ID，并将路由表 ID 替换为实际的 ID。

```
# Get the prefix list ID for S3 in your region
S3_PREFIX_LIST_ID=$(aws ec2 describe-prefix-lists --filters "Name=prefix-
list-name,Values=com.amazonaws.$(aws configure get region).s3" --query
 'PrefixLists[0].PrefixListId' --output text)

# Create the VPC endpoint for S3
aws ec2 create-vpc-endpoint \
  --vpc-id vpc-abcd1234 \
  --service-name com.amazonaws.$(aws configure get region).s3 \
  --route-table-ids rtb-efgh1234 rtb-ijkl1234 \
  --tag-specifications 'ResourceType=vpc-endpoint,Tags=[{Key=Name,Value=S3-Endpoint}]'
```

该命令返回的输出包含 VPC 端点 ID。记下此 ID（例如 vpce-abcd1234）。

配置安全组

安全组充当实例的虚拟防火墙以控制入站和出站流量。为负载均衡器创建安全组，以允许来自任何地方的入站 HTTP 流量。将 vpc-abcd1234 替换为实际的 VPC ID。

```
# Create security group for the load balancer
aws ec2 create-security-group \
  --group-name LoadBalancerSG \
  --description "Security group for the load balancer" \
  --vpc-id vpc-abcd1234 \
  --tag-specifications 'ResourceType=security-
group,Tags=[{Key=Name,Value=LoadBalancerSG}]'
```

该命令返回的输出包含安全组 ID。记下此 ID（例如 sg-abcd1234）。

允许入站 HTTP 流量流向负载均衡器。将 sg-abcd1234 替换为实际的负载均衡器安全组 ID。

```
# Allow inbound HTTP traffic from anywhere
```

```
aws ec2 authorize-security-group-ingress \
  --group-id sg-abcd1234 \
  --protocol tcp \
  --port 80 \
  --cidr 0.0.0.0/0
```

为应用程序服务器创建安全组，仅允许来自负载均衡器的入站流量。将 vpc-abcd1234 替换为实际的 VPC ID。

```
# Create security group for the application servers
aws ec2 create-security-group \
  --group-name AppServerSG \
  --description "Security group for the application servers" \
  --vpc-id vpc-abcd1234 \
  --tag-specifications 'ResourceType=security-
group,Tags=[{Key=Name,Value=AppServerSG}]'
```

该命令返回的输出包含安全组 ID。记下此 ID（例如 sg-efgh1234）。

允许入站 HTTP 流量从负载均衡器安全组流向应用程序服务器。将 sg-efgh1234 替换为实际的应用程序服务器安全组 ID，并将 sg-abcd1234 替换为实际的负载均衡器安全组 ID。

```
# Allow inbound HTTP traffic from the load balancer security group
aws ec2 authorize-security-group-ingress \
  --group-id sg-efgh1234 \
  --protocol tcp \
  --port 80 \
  --source-group sg-abcd1234
```

为 EC2 实例创建启动模板

启动模板包含启动实例的配置信息，例如 AMI ID、实例类型和安全组。首先，创建一个将在实例启动时执行的用户数据脚本。

```
cat > user-data.sh << 'EOF'
#!/bin/bash
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
echo "<h1>Hello from $(hostname -f) in $(curl -s http://169.254.169.254/latest/meta-
data/placement/availability-zone)</h1>" > /var/www/html/index.html
```

```
EOF
```

以 Base64 格式为用户数据脚本编码。

```
USER_DATA=$(base64 -w 0 user-data.sh)
```

查看最新的 Amazon Linux 2 AMI ID。

```
# Get the latest Amazon Linux 2 AMI ID
aws ec2 describe-images --owners amazon --filters "Name=name,Values=amzn2-ami-hvm-
*-x86_64-gp2" "Name=state,Values=available" --query 'sort_by(Images, &CreationDate)
[-1].ImageId' --output text
```

创建启动模板，其中包含 AMI ID、实例类型、安全组和用户数据。将 sg-efgh1234 替换为实际的应用程序服务器安全组 ID，并将 $AMI_ID 和 $USER_DATA 替换为从之前命令获取的值。

```
# Create a launch template
aws ec2 create-launch-template \
  --launch-template-name AppServerTemplate \
  --version-description "Initial version" \
  --tag-specifications 'ResourceType=launch-
template,Tags=[{Key=Name,Value=AppServerTemplate}]' \
  --launch-template-data '{
    "NetworkInterfaces": [{
      "DeviceIndex": 0,
      "Groups": ["sg-efgh1234"],
      "DeleteOnTermination": true
    }],
    "ImageId": "ami-abcd1234",
    "InstanceType": "t3.micro",
    "UserData":
 "IyEvYmluL2Jhc2gKeXVtIHVwZGF0ZSAteQp5dW0gaW5zdGFsbCAteSBodHRwZApzeXN0ZW1jdGwgc3RhcnQgaHR0cGQKc
+SGVsbG8gZnJvbSAkKGhvc3RuYW1lIC1mKSBpbiAkKGN1cmwgLXMgaHR0cDovLzE2OS4yNTQuMTY5LjI1NC9sYXRlc3QvbW
    "TagSpecifications": [{
      "ResourceType": "instance",
      "Tags": [{
        "Key": "Name",
        "Value": "AppServer"
      }]
    }]
  }'
```

创建负载均衡器和目标组

目标组使用指定的协议和端口，将请求路由到已注册的目标（例如 EC2 实例）。为应用程序服务器创建目标组。将 vpc-abcd1234 替换为实际的 VPC ID。

```
# Create a target group
aws elbv2 create-target-group \
  --name AppTargetGroup \
  --protocol HTTP \
  --port 80 \
  --vpc-id vpc-abcd1234 \
  --target-type instance \
  --health-check-protocol HTTP \
  --health-check-path / \
  --health-check-port traffic-port
```

该命令返回的输出包含目标组 ARN。记下此 ARN（例如 arn:aws:elasticloadbalancing:us-east-1:123456789012:targetgroup/AppTargetGroup/abcd1234）。

在公有子网中创建应用程序负载均衡器。将子网 ID 和安全组 ID 替换为实际的 ID。

```
# Create a load balancer
aws elbv2 create-load-balancer \
  --name AppLoadBalancer \
  --subnets subnet-abcd1234 subnet-efgh1234 \
  --security-groups sg-abcd1234 \
  --tags Key=Name,Value=AppLoadBalancer
```

该命令返回的输出包含负载均衡器 ARN。记下此 ARN（例如 arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/app/AppLoadBalancer/abcd1234）。

等待负载均衡器启用后，再继续操作。将 arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/app/AppLoadBalancer/abcd1234 替换为实际的负载均衡器 ARN。

```
# Wait for load balancer to be active
aws elbv2 wait load-balancer-available \
  --load-balancer-arns arn:aws:elasticloadbalancing:us-
east-1:123456789012:loadbalancer/app/AppLoadBalancer/abcd1234
```

为负载均衡器创建侦听器，以将 HTTP 流量转发到目标组。将负载均衡器 ARN 和目标组 ARN 替换为实际的 ARN。

```
# Create a listener
aws elbv2 create-listener \
  --load-balancer-arn arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/
app/AppLoadBalancer/abcd1234 \
  --protocol HTTP \
  --port 80 \
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
east-1:123456789012:targetgroup/AppTargetGroup/abcd1234
```

创建 自动扩缩组

自动扩缩组由一组 Amazon EC2 实例组成，这是实例视为一个逻辑组以实现自动扩缩和管理。创建一个使用启动模板并将实例放置在私有子网中的自动扩缩组。将子网 ID 和目标组 ARN 替换为实际的 ID 和 ARN。

```
# Create an Auto Scaling group
aws autoscaling create-auto-scaling-group \
  --auto-scaling-group-name AppAutoScalingGroup \
  --launch-template LaunchTemplateName=AppServerTemplate,Version='$Latest' \
  --min-size 2 \
  --max-size 4 \
  --desired-capacity 2 \
  --vpc-zone-identifier "subnet-abcd5678,subnet-efgh5678" \
  --target-group-arns arn:aws:elasticloadbalancing:us-east-1:123456789012:targetgroup/
AppTargetGroup/abcd1234 \
  --health-check-type ELB \
  --health-check-grace-period 300 \
  --tags Key=Name,Value=AppServer,PropagateAtLaunch=true
```

测试配置

自动扩缩组启动实例并通过运行状况检查后，即可测试负载均衡器。获取负载均衡器的 DNS 名称。将负载均衡器 ARN 替换为实际的 ARN。

```
# Get the DNS name of the load balancer
aws elbv2 describe-load-balancers \
  --load-balancer-arns arn:aws:elasticloadbalancing:us-
east-1:123456789012:loadbalancer/app/AppLoadBalancer/abcd1234 \
```

```
  --query 'LoadBalancers[0].DNSName' \
  --output text)
```

使用 curl 通过负载均衡器名称测试应用程序。

```
curl http://LoadBalancerName
```

如果您多次刷新页面，应该会看到来自不同可用区中不同实例的响应。

清理资源

完成本教程后，请删除所有资源以避免产生费用。将所有 ID 替换为实际的资源 ID。

```
# Delete the Auto Scaling group
aws autoscaling delete-auto-scaling-group \
  --auto-scaling-group-name AppAutoScalingGroup \
  --force-delete

# Wait for the Auto Scaling group to be deleted
sleep 60

# Delete the load balancer
aws elbv2 delete-load-balancer \
  --load-balancer-arn arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/
app/AppLoadBalancer/abcd1234

# Wait for the load balancer to be deleted
sleep 30

# Delete the target group
aws elbv2 delete-target-group \
  --target-group-arn arn:aws:elasticloadbalancing:us-east-1:123456789012:targetgroup/
AppTargetGroup/abcd1234

# Delete the launch template
aws ec2 delete-launch-template \
  --launch-template-name AppServerTemplate

# Delete the NAT Gateways
aws ec2 delete-nat-gateway --nat-gateway-id nat-abcd1234
aws ec2 delete-nat-gateway --nat-gateway-id nat-efgh1234

# Wait for the NAT Gateways to be deleted
```

```
sleep 90

# Release the Elastic IPs
aws ec2 release-address --allocation-id eipalloc-abcd1234
aws ec2 release-address --allocation-id eipalloc-efgh1234

# Delete the VPC endpoint
aws ec2 delete-vpc-endpoints --vpc-endpoint-ids vpce-abcd1234

# Wait for security group dependencies to clear
sleep 30

# Delete the security groups
aws ec2 delete-security-group --group-id sg-efgh1234
aws ec2 delete-security-group --group-id sg-abcd1234

# Detach the Internet Gateway
aws ec2 detach-internet-gateway --internet-gateway-id igw-abcd1234 --vpc-id vpc-
abcd1234

# Delete the Internet Gateway
aws ec2 delete-internet-gateway --internet-gateway-id igw-abcd1234

# Delete the route tables
aws ec2 delete-route-table --route-table-id rtb-efgh1234
aws ec2 delete-route-table --route-table-id rtb-ijkl1234
aws ec2 delete-route-table --route-table-id rtb-abcd1234

# Delete the subnets
aws ec2 delete-subnet --subnet-id subnet-abcd1234
aws ec2 delete-subnet --subnet-id subnet-efgh1234
aws ec2 delete-subnet --subnet-id subnet-abcd5678
aws ec2 delete-subnet --subnet-id subnet-efgh5678

# Delete the VPC
aws ec2 delete-vpc --vpc-id vpc-abcd1234
```

后续步骤

现在，您已经创建包含私有子网和 NAT 网关的 VPC，可能还想探索以下相关主题：

- VPC 的安全最佳实践
- 使用 VPC 流日志记录 IP 流量

- 自动扩缩组扩缩策略
- 负载均衡器目标组运行状况检查

Amazon VPC 配额

以下表格列出了您的 AWS 账户的 Amazon VPC 资源限额（之前称为限制）。除非另外指明，否则这些配额是针对每个区域的。

如果您请求对每个资源提升适用的配额，我们将提升该区域中所有资源的配额。

VPC 和子网

名称	默认值	可调整	评论
每个区域的 VPC 数	5	是	提高该配额会使每个区域的互联网网关数量配额提高同样的数量。 您可以提高此限制，以便在每个区域拥有数百个 VPC。
每个 VPC 的子网数量	200	是	
每个 VPC 的 IPv4 CIDR 块数	5	是 （最多 50）	此主 CIDR 数据块和所有辅助 CIDR 数据块计数计入此配额中。
每个 VPC 的 IPv6 CIDR 块数	5	是 （最多 50）	您可以分配给单个 VPC 的 CIDR 数量。
每个区域每个账户的 VPC 屏蔽公共访问权限排除项数	50	是。要请求增加，请使用 AWS Support Center Console创建提高服务配额的工单。	您可以在账户中创建的 VPC BPA 排除项的数量。

DNS

每个 EC2 实例可以每秒为每个网络接口向 Route 53 Resolver（具体指 .2 地址，例如，10.0.0.2 和 169.254.169.253）发送 1024 个数据包。无法提高此配额。由 Route 53 Resolver 支持的每秒 DNS 查询数量因查询类型、响应大小和所用协议而异。有关可扩展 DNS 架构的更多信息和建议，请参阅具有 Active Directory 的 AWS 混合 DNS 技术指南。

弹性 IP 地址

名称	默认值	可调整	评论
每个区域的弹性 IP 地址数量	5	是	此限额适用于单个 AWS 账户 VPC 和共享 VPC。
每个公有 NAT 网关的弹性 IP 地址	2	可以	您可以请求将限额提高到 8。

网关

名称	默认值	可调整	评论
每个区域的仅出口互联网网关数	5	是	要提高此限额，请提高每个区域的 VPC 限额。 您一次只能将一个仅出口互联网网关连接到 VPC。
每个区域的互联网网关数	5	是	要提高此限额，请提高每个区域的 VPC 限额。 您一次只能将一个互联网网关连接到 VPC。
每个可用区的 NAT 网关	5	是	NAT 网关仅计入 `pending`、`active` 和 `deleting` 状态的限额。

名称	默认值	可调整	评论
每个 NAT 网关的私有 IP 地址限额	8	是	
每个 VPC 的运营商网关数	1	否	

客户管理的前缀列表

尽管可以调整客户自行管理的前缀列表的默认限额，但您无法使用服务配额控制台调整配额。您必须使用 AWS Support Center Console 创建提高服务配额的工单。

名称	默认值	可调整	评论
每个区域的前缀列表数	100	是	
每个前缀列表的版本数	1000	是	如果前缀列表中有 1000 个存储版本，并且您添加了新版本，将移除最旧的版本以便添加新版本。
每个前缀列表的最大条目数	1000	是	您可以将客户托管的前缀列表的最大数量调整为 1000。有关更多信息，请参阅 调整前缀列表的大小。当您在资源中引用前缀列表时，前缀列表的最大条目数占用资源的条目数限额。例如，如果您创建一个包含最多 20 个条目的前缀列表，并且在安全组规则中引用该前缀列表，这将视为 20 个安全组规则。
对每种资源类型的前缀列表的引用	10000	是	此配额按可引用前缀列表的资源类型应用。例如，您可以在所有安全组中具有 10000 个对前缀列表的引用，并在所有子网路由表中具有 10000 个对前缀列表的引用。如果您与其他 AWS 账户共享前缀列表，则其他账户对您的前缀列表的引用将计入您的配额。

网络 ACL

名称	默认值	可调整	评论
每个 VPC 的网络 ACL 数	200	是	在 VPC 中，您可以将一个网络 ACL 关联到一个或多个子网。
每个网络 ACL 的规则数	20	是	此限额确定入站规则的最大数量和出站规则的最大数量。此限额最多可以增加到 40 个入站规则和 40 个出站规则（总共 80 个规则），但网络性能可能会受到影响。

网络接口

名称	默认值	可调整	评论
每个实例的网络接口	因实例类型而异	否	有关更多信息，请参阅每个实例类型的网络接口。
每个区域的网络接口数	5000	是	此限额适用于单个 AWS 账户 VPC 和共享 VPC。此限制针对每个可用区（AZ）强制执行。例如，如果网络接口在三个可用区中，则每个可用区的限制为 5,000，区域的限制为 15,000。

路由表

名称	默认值	可调整	评论
每个 VPC 的路由表数	200	是	主路由表计入此配额。请注意，如果您请求增加路由表的配额，则可能还需要请求增加子网的配额。虽然路由表可以与多个子网分享，但是只能为一个子网关联一个路由表。

名称	默认值	可调整	评论
每个路由表的路由 (非传播路由)	500	是	可以将该配额提高至最大值 1000；但是，网络性能可能会受到影响。将单独为 IPv4 路由和 IPv6 路由实施该配额。 如果您有 125 个以上的路由，我们建议您对调用进行分页以描述路由表，从而获得更好的性能。
每个路由表传播的路由	100	否	如果您需要其他前缀，请通告原定设置路由。

路由服务器

名称	默认值	可调整	评论
每个 VPC 的路由服务器数	5	可以。要请求增加，请使用 AWS Support Center Console创建提高服务配额的工单。	
每个路由服务器的路由服务器端点数	10	是。要请求增加，请使用 AWS Support Center Console创建提高服	

名称	默认值	可调整	评论
		务配额的工单。	
每个网络接口的对等连接会话数	20	是。要请求增加，请使用 AWS Support Center Console创建提高服务配额的工单。	
每个路由服务器和子网的路由服务器端点数	2	否	为实现冗余，同一路由服务器在同一个子网中只能有两个端点。
每个路由服务器对等的路由数	100	否	此项是可通过路由服务器对等动态通告的路由数量
每个路由服务器的路由数	100	否	此项是可以在路由服务器的转发信息库（FIB）中安装的路由数量。

安全组

名称	默认值	可调整	评论
每个区域的 VPC 安全组	2,500	是	此限额适用于单个 AWS 账户 VPC 和共享 VPC。 如果您将此配额增加到一个区域中超过 5000 个安全组，我们建议您对调用进行分页以描述安全组，从而获得更好的性能。

名称	默认值	可调整	评论
每个安全组的入站或出站规则	60	是	将单独为入站和出站规则实施此配额。对于默认限额为 60 条规则的账户，安全组可以设置 60 条入站规则和 60 条出站规则。次卧，将单独为 IPv4 规则和 IPv6 规则实施此配额。对于默认限额为 60 条规则的账户，安全组可以分别为 IPv4 流量和 IPv6 流量设置 60 条入站规则。有关更多信息，请参阅 the section called “安全组大小”。 配额更改适用于入站和出站规则。该配额值与每个网络接口的安全组配额值的积不得超过 1000。
每个网络接口的安全组数	5	是 （最多 16）	该配额值乘以每个安全组的规则配额值的积不得超过 1000。

VPC 子网共享

所有标准 VPC 配额均适用于共享的 VPC 子网。

名称	默认值	可调整	评论
每个 VPC 的参与者账户	100	是	可以与其共享 VPC 中的子网的不同参与者账户的最大数量。这是每个 VPC 的配额，并应用于 VPC 中共享的所有子网。 VPC 拥有者可以查看附加到参与者资源的网络接口和安全组。
可以与账户共享的子网	100	是	这是可以与 AWS 账户共享的最大子网数量。

网络地址用量

网络地址用量（NAU）由托管式前缀列表中的 IP 地址、网络接口和 CIDR 组成。NAU 是应用于 VPC 中资源的指标，可以帮助您规划和监控 VPC 的大小。有关更多信息，请参阅 网络地址用量。

构成 NAU 计数的资源的服务配额各有不同。即使 VPC 具有可用的 NAU 容量，但如果资源已超过其服务配额，您也无法在 VPC 内启动资源。

名称	默认值	可调整	评论
网络地址用量	64,000	是（最多 256,000）	每个 VPC 的最大 NAU 单元数。
对等网络地址用量	128,000	是（最多 512,000）	VPC 及其所有区域内对等 VPC 的最大 NAU 单元数。跨不同区域的对等 VPC 不会计入此数量。

Amazon EC2 API 节流

有关 Amazon EC2 节流的信息，请参阅《Amazon EC2 开发人员指南》中的 Request throttling。

其他配额资源

有关更多信息，请参阅下列内容：

- AWS Client VPN 管理员指南中的 AWS Client VPN 配额
- AWS Direct Connect 用户指南中的 AWS Direct Connect 配额
- Amazon VPC 对等连接指南中的对等配额
- AWS PrivateLink 指南中的 PrivateLink 配额
- AWS Site-to-Site VPN 用户指南中的 Site-to-Site VPN 配额
- Amazon VPC Traffic Mirroring 指南中的流量镜像配额
- Amazon VPC Transit 网关指南中的 Transit 网关配额

文档历史记录

下表介绍了每个《Amazon VPC 用户指南》版本中的重要更改。

变更	说明	日期
使用 Amazon VPC Route Server 在 VPC 中动态路由	Amazon VPC Route Server 简化了部署在 VPC 内的工作负载与其互联网网关之间的流量路由。借助此功能，VPC Route Server 使用首选的 IPv4 或 IPv6 路由动态更新 VPC 和网关路由表，以实现对这些工作负载的路由容错能力。这使您能够自动重新路由 VPC 内的流量，从而提高 VPC 路由的可管理性，以及与第三方工作负载的互操作性。	2025 年 3 月 31 日
AWS 托管策略更新	Amazon VPC 更新了 AmazonVPCFullAccess 和 AmazonVPCReadOnlyAccess 托管策略。	2024 年 12 月 9 日
VPC BPA 的声明式策略支持	如果您使用 AWS Organizations 来管理组织中的账户，则可以使用声明式策略对组织中的账户强制执行 VPC BPA。	2024 年 12 月 1 日
VPC 屏蔽公共访问权限（BPA）	通过 VPC 屏蔽公共访问权限（BPA），您可以阻止您在区域中拥有的 VPC 和子网中的资源通过互联网网关和仅出口互联网网关进行访问或从互联网进行访问。	2024 年 11 月 19 日

共享安全组	此功能使您能够与其他 AWS Organizations 账户共享安全组。	2024 年 10 月 30 日
安全组 VPC 关联	此功能使您能够将一个安全组与同一区域中的多个 VPC 关联。	2024 年 10 月 30 日
NAT 网关 MTU 支持	NAT 网关可支持最大传输单位（MTU）为 8500 的流量。	2024 年 9 月 10 日
私有 IPv6 寻址	添加了关于私有 IPv6 寻址的信息。私有 IPv6 地址仅在 Amazon VPC IP 地址管理器中可用。	2024 年 8 月 8 日
IPv6 首选租赁时间	现在，您可以选择为其分配了 IPv6 的正在运行的实例续订 DHCPv6 租约的频率。	2024 年 2 月 20 日
指南结构审查和改进	我们对指南的结构进行了审查和改进，可改善与查找特定场景信息相关的客户体验。	2024 年 2 月 20 日
AWS 托管策略更新	Amazon VPC 更新了 AmazonVPCFullAccess 和 AmazonVPCReadOnlyAccess 托管策略。	2024 年 2 月 8 日
AWS 托管策略更新	Amazon VPC 更新了 AmazonVPCCrossAccountNetworkInterfaceOperations 托管策略。	2023 年 9 月 25 日

EC2-Classic 已弃用	通过使用 EC2-Classic，EC2 实例会在一个可与其他客户共享的扁平化网络中运行。Amazon VPC 将替代 EC2-Classic。通过使用 Amazon VPC，实例会在一个逻辑上与 AWS 账户 分离的虚拟私有云（VPC）中运行。	2023 年 7 月 31 日
向 NAT 网关添加辅助 IPv4 地址	您可以向公有和私有 NAT 网关添加辅助私有 IPv4 地址。辅助 IPv4 地址增加了可用的端口数量，从而提高了工作负载使用 NAT 网关建立连接的并发连接数限制。	2023 年 1 月 31 日
遵循 IAM 最佳实践	更新了指南，使其符合 IAM 最佳实践。有关更多信息，请参阅 IAM 安全最佳实践。	2023 年 1 月 4 日
选择 NAT 网关的私有 IP 地址	现在，您创建 NAT 网关时，可以选择分配给 NAT 网关的私有 IP 地址。以前，会从子网的 IP 地址范围自动分配私有 IP 地址。	2022 年 11 月 17 日
IPv6 默认网关路由器配置	现在，预留了三个 IPv6 地址供默认 VPC 路由器使用。	2022 年 11 月 11 日
转移弹性 IP 地址	您现在可以将弹性 IP 地址从一个 AWS 账户转移到另一个账户。	2022 年 10 月 31 日
网络地址用量指标	您可以为 VPC 启用网络地址用量指标，以帮助您规划和监控 VPC 的大小。	2022 年 10 月 4 日

将流日志发布到 Amazon Data Firehose	您可以将 Amazon Data Firehose 传输流指定为流日志数据的目的地。	2022 年 9 月 8 日
NAT 网关带宽	NAT 网关现在支持高达 100 Gbps 的带宽（以前为 45 Gbps），每秒可处理多达 1 千万个数据包（以前为 4 百万个数据包）。	2022 年 6 月 15 日
多个 IPv6 CIDR 块	您可以向 VPC 关联最多 5 个 IPv6 CIDR 块。	2022 年 5 月 12 日
重新企业	本 Amazon Virtual Private Cloud 用户指南的全面重组。	2022 年 1 月 2 日
NAT 网关 IPv6 到 IPv4	NAT 网关支持从 IPv6 到 IPv4 的网络地址转换，这通常称为 NAT64。	2021 年 11 月 24 日
VPC 中的仅 IPv6 子网	您可以创建仅 IPv6 子网，可在其中启动仅 IPv6 的 EC2 实例。	2021 年 11 月 23 日
VPC 流日志将选项发送到 Amazon S3	您可以指定 Apache Pparquet 日志文件格式、每小时分区和 HIVE 兼容 S3 前缀。	2021 年 10 月 13 日
Amazon EC2 全局视图	通过 Amazon EC2 全局视图，您可以在单个控制台中查看跨多个 AWS 区域的 VPC、子网、实例、安全组和卷。	2021 年 9 月 1 日

更具体的路由	您可以在您的路由表中添加比本地路由更具体的路由。您可以使用更具体的路由将 VPC（东-西流量）内子网之间的流量重新导向到中间盒设备。您可以设置路由的目的地来匹配 VPC 中子网的整个 IPv4 或 IPv6 CIDR 块。	2021 年 8 月 30 日
针对安全组规则的资源 ID 和标记支持	您可以按资源 ID 引用安全组规则。您还可以为安全组规则添加标签。	2021 年 7 月 7 日
私有 NAT 网关	您可以使用私有 NAT 网关，在 VPC 之间或 VPC 与本地网络之间实现仅出站私有通信。	2021 年 6 月 10 日
在创建时添加标签	您可以在创建 VPC、DHCP 选项、互联网网关、仅出口网关、网络 ACL 和安全组时添加标签。	2020 年 6 月 30 日
托管前缀列表	您可以在前缀列表中创建和管理一组 CIDR 块。	2020 年 6 月 29 日
流日志增强功能	新的流日志字段可用，您可以为发布到 CloudWatch Logs 的流日志指定自定义格式。	2020 年 5 月 4 日
对流日志的标记支持	您可以向流日志添加标签。	2020 年 3 月 16 日
有关 NAT 网关创建的标签	您可以在创建 NAT 网关时添加标签。	2020 年 3 月 9 日
流日志的最大聚合时间间隔	您可以指定捕获流并聚合到流日志记录中的最长时间段。	2020 年 2 月 4 日

网络边界组配置	您可以从 Amazon Virtual Private Cloud Console 为 VPC 配置网络边界组。	2020 年 1 月 22 日
网关路由表	您可以将路由表与网关相关联，并将入站 VPC 流量路由到 VPC 中的特定网络接口。	2019 年 12 月 3 日
流日志增强功能	您可以为流日志指定自定义格式并选择在流日志记录中返回哪些字段。	2019 年 9 月 11 日
VPC 共享	您可以与同一AWS企业中的多个账户共享位于同一 VPC 中的子网。	2018 年 11 月 27 日
创建默认子网	您可以在没有默认子网的可用区中创建一个默认子网。	2017 年 11 月 9 日
NAT 网关的标记支持	您可以给自己的 NAT 网关加标签。	2017 年 9 月 7 日
适用于 NAT 网关的 Amazon CloudWatch 指标	您可以查看适用于 NAT 网关的 CloudWatch 指标。	2017 年 9 月 7 日
安全组规则说明	您可以向安全组规则添加说明。	2017 年 8 月 31 日
VPC 的辅助 IPv4 CIDR 块	您可以向 VPC 中添加多个 IPv4 CIDR 块。	2017 年 29 月 8 日
恢复弹性 IP 地址	如果您释放了一个弹性 IP 地址，则可能能够恢复它。	2017 年 8 月 11 日
创建默认 VPC	如果您删除了现有默认 VPC，则可以创建一个新的默认 VPC。	2017 年 27 月 7 日

IPv6 支持	您可以将一个 IPv6 CIDR 块与您的 VPC 关联并为您的 VPC 中的资源分配 IPv6 地址。	2016 年 12 月 1 日
对非 RFC 1918 IP 地址范围的 DNS 解析支持	Amazon DNS 服务器现在可以将私有 DNS 主机名解析为全部地址空间内的私有 IP 地址。	2016 年 10 月 24 日
NAT 网关	您可在公有子网中创建 NAT 网关，并让私有子网中的实例向 Internet 或其他 AWS 服务发出出站流量。	2015 年 12 月 17 日
VPC 流日志	您可以创建流日志以捕获有关传入和传出您的 VPC 中的网络接口的 IP 流量的信息。	2015 年 6 月 10 日
ClassicLink	您可以使用 ClassicLink 将 EC2-Classic 实例链接到您账户中的 VPC。您可以将 VPC 安全组与 EC2-Classic 实例关联起来，从而允许 EC2-Classic 实例与 VPC 中使用私有 IP 地址的实例进行通信。	2015 年 1 月 7 日
使用私有托管区域	您可以使用在 Route 53 中的私有托管区域中定义的自定义 DNS 域名访问您的 VPC 中的资源。	2014 年 11 月 5 日
修改子网的公有 IP 寻址属性	您可以修改子网的公有 IP 寻址属性以指示在该子网中启动的实例是否应接收公有 IP 地址。	2014 年 6 月 21 日
分配公有 IP 地址	您可以在启动期间向实例分配公有 IP 地址。	2013 年 8 月 20 日

启用 DNS 主机名称并禁用 DNS 解析	您可以修改 VPC 默认值、禁用 DNS 解析并启用 DNS 主机名。	2013 年 3 月 11 日
VPC 无处不在	增加了对五个AWS区域的 VPC、多个可用区中的 VPC、每个AWS账户多个 VPC 以及每个 VPC 多个 VPN 连接的支持。	2011 年 8 月 3 日
专用实例	专用实例是在您的 VPC 中启动、运行单个客户专用硬件的 Amazon EC2 实例。	2011 年 3 月 27 日