# 在公共云中为 ASA Virtual 部署 ASA 集群

通过集群，您可以将多台 ASA Virtual 组合成一个逻辑设备。集群具有单个设备的全部便捷性（管理、集成到一个网络中），同时还能实现吞吐量增加和多个设备的冗余性。您可以使用以下方法在公共云中部署 ASA Virtual 集群：

- 的 Amazon Web Services (AWS)
- 的 Microsoft Azure

目前仅支持路由防火墙模式。

**注释** 使用集群时，有些功能不受支持。请参阅集群不支持的功能，第 75 页。


- 关于公共云中的 ASA Virtual 集群，第 1 页
- ASA Virtual 集群许可证，第 5 页
- ASA Virtual 集群要求和前提条件，第 6 页
- ASA Virtual 集群准则，第 7 页
- 在 AWS 中部署集群，第 8 页
- 在 Azure 中部署集群，第 26 页
- ASA Virtual Azure 上的集群自动扩展解决方案，第 37 页
- 自定义集群操作，第 55 页
- 管理集群节点，第 59 页
- 监控集群，第 63 页
- 集群参考，第 75 页
- 关于公共云中的 ASA Virtual 集群的历史记录，第 89 页


# 关于公共云中的 ASA Virtual 集群

本节介绍集群架构及其工作原理。

# 集群如何融入网络中

集群包含多台防火墙，作为单一设备工作。要用作集群，该防火墙需要以下基础设施：

- 独立的网络（称为集群控制链路），通过 VXLAN 接口用于集群内的通信。VXLAN 充当第 3 层物理网络上的第 2 层虚拟网络，让 ASA Virtual 能够通过集群控制链路发送广播/组播消息。
- 负载均衡器 - 对于外部负载均衡，您有以下选择：
  - AWS 网关负载均衡器

    AWS 网关负载均衡器结合了透明网络网关和按需分配流量和扩展虚拟设备的负载均衡器。ASA Virtual 支持使用 Geneve 接口单臂代理且具有分布式数据平面的网关负载均衡器集中控制平面（网关负载均衡器终端）。
  - 使用内部和外部路由器（例如思科云服务路由器）的等价多路径路由 (ECMP)

    ECMP 路由可以通过路由指标并列最优的多条“最佳路径”转发数据包。它与 EtherChannel 一样，也可以使用源和目标 IP 地址及/或源和目标端口的散列值将数据包发送到下一跃点。如果将静态路由用于 ECMP 路由，则 ASA Virtual 故障会导致问题；如果继续使用该路由，发往故障 ASA Virtual 的流量将丢失。因此，如果使用静态路由，请务必要使用静态路由监控功能，例如对象跟踪。我们还建议使用动态路由协议来添加和删除路由，在这种情况下，您必须配置每台 ASA Virtual 使之加入动态路由。

**注释** 负载均衡不支持第 2 层跨区以太网通道。

# 集群节点

集群节点协调工作来实现安全策略和流量的共享。本节介绍每种节点角色的性质。

## 引导程序配置

您要在每台设备上配置最低的引导程序配置，包括集群名称、集群控制链路接口和其他集群设置。启用集群的第一个节点通常成为控制节点。在后续节点上启用集群时，这些设备将作为数据节点加入集群。

## 控制和数据节点角色

一个集群成员是控制节点。如果多个集群节点同时上线，则控制节点由引导程序配置中的优先级设置决定；优先级可设置为 1 到 100，其中 1 为最高优先级。所有其他成员都是数据节点。 通常，当您首次创建集群时，您添加的第一个节点会成为控制节点，因为它是到目前为止集群中的唯一节点。

必须只能在控制节点上执行所有配置（引导程序配置除外）；然后配置将被复制到数据节点中。如果是接口等物理资产，控制节点的配置将被镜像到所有数据节点。例如，如果将以太网接口 1/2 配置为内部接口，将以太网接口 1/1 配置为外部接口，则这些接口也将在数据节点上用作内部和外部接口。

有些功能在集群中无法扩展，控制节点将处理这些功能的所有流量。

# 独立接口

您可以将集群接口配置为独立接口。

独立接口是正常的路由接口，每个接口都有自己的本地 IP 地址。必须仅在控制节点上配置接口配置，并且每个接口都要使用 DHCP。

注释 不支持第 2 层跨区以太网通道。

# 集群控制链路

每个节点必须将一个接口作为集群控制链路的 VXLAN (VTEP) 接口。 有关 VXLAN 的详细信息，请参阅VXLAN 接口。

### VXLAN 隧道端点

VXLAN 隧道终端 (VTEP) 设备执行 VXLAN 封装和解封。每个 VTEP 有两种接口类型：一个或多个虚拟接口称为 VXLAN 网络标识符 (VNI) 接口；以及称为 VTEP 源接口的常规接口，用于为 VTEP 之间的 VNI 接口建立隧道。VTEP 源接口连接到传输 IP 网络，进行 VTEP 至 VTEP 通信。

### VTEP 源接口

VTEP 源接口是一个计划要将其与 VNI 接口相关联的常规ASA virtual 接口。您可以将一个 VTEP 源接口配置为集群控制链路。源接口会被保留，以便仅供集群控制链路使用。每个 VTEP 源接口在同一子网上都有一个 IP 地址。此子网应与所有其他流量隔离，并且只包括集群控制链路接口。

### VNI 接口

VNI 接口类似于 VLAN 接口：它是一个虚拟接口，通过使用标记，实现网络流量在给定物理接口上的分离。您只能配置一个 VNI 接口。每个 VNI 接口在同一子网上都有一个 IP 地址。

### 对等体 VTEP

与数据接口的常规 VXLAN 只允许单个 VTEP 对等体不同，ASA virtual 集群允许您配置多个对等体。

## 集群控制链路流量概述

集群控制链路流量包括控制流量和数据流量。

控制流量包括：

- 控制节点选举。
- 配置复制。

- 运行状况监控。

数据流量包括：

- 状态复制。
- 连接所有权查询和数据包转发。

### 集群控制链路故障

如果某台设备的集群控制链路线路协议关闭，则集群将被禁用；数据接口关闭。当您修复集群控制链路之后，必须通过重新启用集群来手动重新加入集群。

注释 当 ASA Virtual 处于非活动状态时，所有数据接口关闭；只有管理专用接口可以发送和接收流量。管理接口将保持打开，使用设备从 DHCP 或集群 IP 池接收的 IP 地址。如果使用集群 IP池，在重新加载而设备在集群中仍然处于非活动状态时，则管理接口将无法访问（因为它届时将使用与控制节点相同的主 IP 地址）。您必须使用控制台端口（如果可用）来进行任何进一步配置。

## 配置复制

集群中的所有节点共享一个配置。您只能在控制节点上进行配置更改（引导程序配置除外），这些更改会自动同步到集群中的所有其他节点。

# ASA Virtual集群管理

使用 ASA Virtual集群的一个好处可以简化管理。本节介绍如何管理集群。

### 管理网络

我们建议将所有节点都连接到一个管理网络。此网络与集群控制链路分隔开来。

### 管理接口

使用 Management 0/0 接口进行管理。

注释 您不能为管理接口启用动态路由。您必须使用静态路由。

您可以使用静态寻址或 DHCP 作为管理 IP 地址。

如果您使用静态寻址，则可以使用集群的主集群 IP 地址是集群的固定地址，而该集群始终属于当前的控制节点。您还要为每个接口配置一个地址范围，以便包括当前控制节点在内的每个节点都可以使用该范围内的本地地址。主集群 IP 地址提供对地址的统一管理访问；当控制节点更改时，主集群 IP 地址将转移到新的控制节点，使集群的管理得以无缝继续。本地 IP 地址用于路由，在排除故障时

也非常有用。例如，您可以连接到主集群 IP 地址来管理集群，该地址始终属于当前的控制节点。要管理单个成员，您可以连接到本地 IP 地址。对于 TFTP 或系统日志等出站管理流量，包括控制节点在内的每个节点都使用本地 IP 地址连接到服务器。

如果使用 DHCP，则不使用本地地址池或主集群 IP 地址。

注释 传入设备的流量必须指向节点的管理 IP 地址；传入设备的流量不会通过群集控制链路转发到任何其他节点。

## 控制节点管理与数据节点管理

所有管理和监控均可在控制节点上进行。从控制节点中，您可以检查所有节点的运行时统计信息、资源使用情况或其他监控信息。您也可以向集群中的所有节点发出命令，并将控制台消息从数据节点复制到控制节点。

如果需要，您可以直接监控数据节点。虽然在控制节点上可以执行文件管理，但在数据节点上也可以如此（包括备份配置和更新映像）。以下功能在控制节点上不可用：

- 监控每个节点的集群特定统计信息。
- 监控每个节点的系统日志（控制台复制启用时发送至控制台的系统日志除外）。
- SNMP
- NetFlow

## 加密密钥复制

当您在控制节点上创建加密密钥时，该密钥将复制到所有数据节点。如果您有连接到主集群 IP 地址的 SSH 会话，则控制节点发生故障时连接将断开。新控制节点对 SSH 连接使用相同的密钥，这样当您重新连接到新的控制节点时，无需更新缓存的 SSH 主机密钥。

## ASDM 连接证书 IP 地址不匹配

默认情况下，ASDM 连接将根据本地 IP 地址使用自签名证书。如果使用 ASDM 连接到主集群 IP 地址，则可能会因证书使用的是本地 IP 地址而不是主集群 IP 地址，系统会显示一则警告消息，指出 IP 地址不匹配。您可以忽略该消息并建立 ASDM 连接。但是，为了避免此类警告，您也可以注册一个包含主集群 IP 地址和 IP 地址池中所有本地 IP 地址的证书。然后，您可将此证书用于每个集群成员。有关详细信息，请参阅 https://www.cisco.com/c/en/us/td/docs/security/asdm/identity-cert/cert-install.html。

# ASA Virtual 集群许可证

每个集群节点都需要相同的模型许可证。我们建议为所有节点使用相同数量的 CPU 和内存，否则将限制所有节点上的性能，以匹配功能最低的成员。吞吐量级别将从控制节点复制到每个数据节点，以便它们匹配。

注释 如果取消注册 ASA Virtual 从而使得其未经许可，则在重新加载 ASA Virtual 后，它将恢复到严格的速率限制状态。未经许可的低性能集群节点将对整个集群的性能产生负面影响。请务必保留所有集群节点的许可，或删除任何未经许可的节点。

# ASA Virtual 集群要求和前提条件

### 型号要求

- ASAv30, ASAv50, ASAv100
- 以下公共云服务：
  - 的 Amazon Web Services (AWS)
  - 的 Microsoft Azure
- 最多 16 个节点

另请参阅《ASA Virtual 入门指南》中有关 ASA Virtual 的一般要求。

### 硬件和软件要求

集群中的所有节点：

- 必须在同一个性能层。我们建议对所有节点都使用相同数量的CPU和内存，否则所有节点上的性能将受到限制，以匹配性能最低的节点。
- 除在映像升级时以外，必须运行完全相同的软件。支持无中断升级。不匹配的软件版本可能会导致性能不佳，因此请务必在同一维护窗口中升级所有节点。
- 支持单个可用性区域部署。
- 集群控制链路接口必须位于同一子网中，因此集群应部署在同一子网中。

### MTU

确保连接到集群控制链路的端口配置了正确（更高）的 MTU。如果存在不匹配的 MTU，则集群形成将失败。当某个节点加入集群时，它会向控制节点发送ping，其数据包大小与集群控制链路MTU匹配，从而检查 MTU 兼容性。如果 ping 失败，系统会生成通知，以便您纠正连接的交换机上 MTU 不匹配的问题，然后重试。

默认情况下，集群控制链路 MTU 会被设置为比数据接口高 154 字节。由于集群控制链路流量包括数据包转发，因此集群控制链路需要能够容纳完整大小的数据包以及集群流量开销（100 字节）加上 VXLAN 开销（54 字节）。

对于具有 GWLB 的 AWS，数据接口使用 Geneve 封装。在这种情况下，整个以太网数据报将被封装，因此，新数据包更大，需要更大的 MTU。您应将源接口 MTU 设置为网络 MTU + 306 字节。因

此，对于标准的 1500 MTU 网络路径，源接口 MTU 应为 1806，而集群控制链路 MTU 应为 +154, 1960。

对于具有 GWLB 的 Azure，数据接口使用 VXLAN 封装。在这种情况下，整个以太网数据报将被封装，因此，新数据包更大，需要更大的 MTU。您应将源接口 MTU 设置为网络 MTU + 54 字节。

下表显示了建议的集群控制链路 MTU 和数据接口 MTU。

**注释** 我们不建议将集群控制链路 MTU 设置为介于 2561 和 8362 之间的值；由于块池处理，此 MTU 大小不是系统运行的最佳值。

**表 *1:* 建议的 *MTU***

| 公共云 | 集群控制链路 MTU | 数据接口 MTU |
|---|---|---|
| 具有 GWLB 的 AWS | 1960 | 1806 |
| AWS | 1654 | 1500 |
| 具有 GWLB 的 Azure | 1554 | 1454 |
| Azure | 1554 | 1400 |

# ASA Virtual 集群准则

### 高可用性

集群不支持高可用性。

### IPv6

集群控制链路只有在使用 IPv4 时才受支持。

### 其他准则

- 当拓扑发生重大更改时（例如添加或删除 EtherChannel 接口、启用或禁用 ASA Virtual 或交换机上的接口、添加额外的交换机形成冗余交换机系统），您应禁用运行状态检查功能，还要禁用对已禁用接口的接口监控。当拓扑更改完成且配置更改已同步到所有设备后，您可以重新启用接口运行状态检查功能。
- 将节点添加到现有集群时或重新加载节点时，会有限地暂时丢弃数据包/断开连接；这是预期的行为。有些时候，丢弃的数据包会挂起连接；例如，丢弃 FTP 连接的 FIN/ACK 数据包将使 FTP 客户端挂起。在此情况下，您需要重新建立 FTP 连接。

- 对于解密的 TLS/SSL 连接，解密状态不同步，如果连接所有者失败，则解密的连接将重置。需要与新节点建立新的连接。未解密的连接（它们匹配“不解密”规则）不受影响，并且可以正确复制。
- 不支持动态扩展。
- 在每个维护窗口完成后执行全局部署。
- 确保不要一次从自动扩展组 (AWS) 或规模集 (Azure) 中删除多个设备。我们还建议您先在设备上运行 **cluster disable** 命令，然后再从组东扩展组 (AWS) 或规模集(Azure) 中删除设备。
- 如果要禁用集群中的数据节点和控制节点，我们建议您在禁用控制节点之前先禁用数据节点。如果在集群中有其他数据节点时禁用了某个控制节点，则必须将其中一个数据节点升级为控制节点。请注意，角色更改可能会对集群造成干扰。
- 在本指南中提供的 Day 0 配置脚本中，您可以根据需要更改 IP 地址，提供自定义接口名称，并更改 CCL-Link 接口的顺序。

**集群默认设置**

- 将自动生成 cLACP 系统 ID 且系统优先级默认为 1。
- 默认情况下，集群运行状况检查功能处于启用状态，保持时间为 3 秒。默认情况下，在所有接口上启用接口运行状况监控。
- 用于发生故障的集群控制链路的集群自动重新加入功能为每 5 分钟尝试无限次。
- 用于发生故障的数据接口的集群自动重新加入功能为每 5 分钟尝试 3 次，增量间隔设置为 2。
- 对于 HTTP 流量，默认启用 5 秒的连接复制延迟。

# 在 AWS 中部署集群

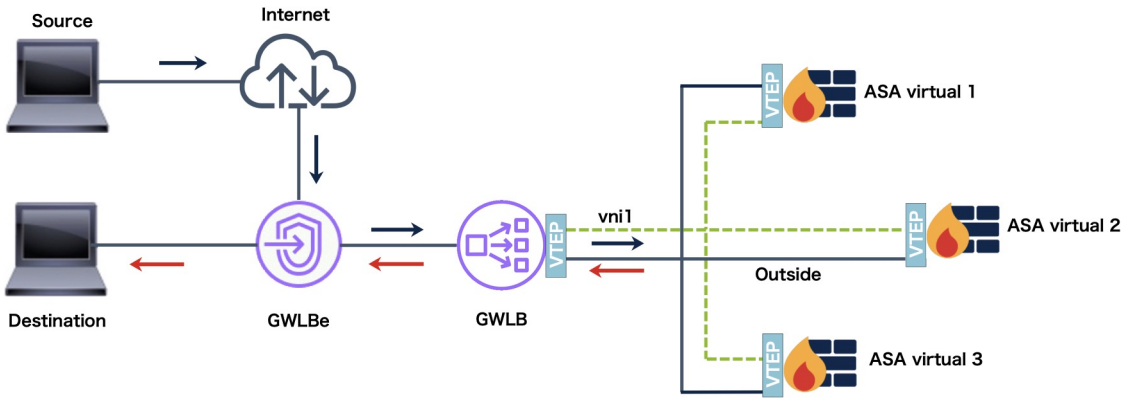
要在 AWS 中部署集群，您可以手动部署或使用 CloudFormation 模板来部署堆栈。您可以将集群与 AWS 网关负载均衡器或非本地负载均衡器（例如思科云服务路由器）配合使用。

## AWS 网关负载均衡器和 Geneve 单臂代理

**注释** 这是 Geneve 接口当前唯一支持的使用案例。

AWS网关负载均衡器结合了透明网络网关和按需分配流量和扩展虚拟设备的负载均衡器。ASA Virtual支持具有分布式数据平面的网关负载均衡器集中控制平面（网关负载均衡器终端）。下图显示了从网关负载均衡器终端转发到网关负载均衡器的流量。网关负载均衡器会在多个ASA Virtual之间进行均衡，这些流量在丢弃流量或将其发送回网关负载均衡器之前对其进行检查（掉头流量）。然后，网关负载均衡器会将流量发送回网关负载均衡器终端和目的地。

图 *1: Geneve* 单臂代理




# 拓扑示例

### ASA Virtual 在 AWS 区域的单个和多个可用性区域中使用自动扩展进行集群

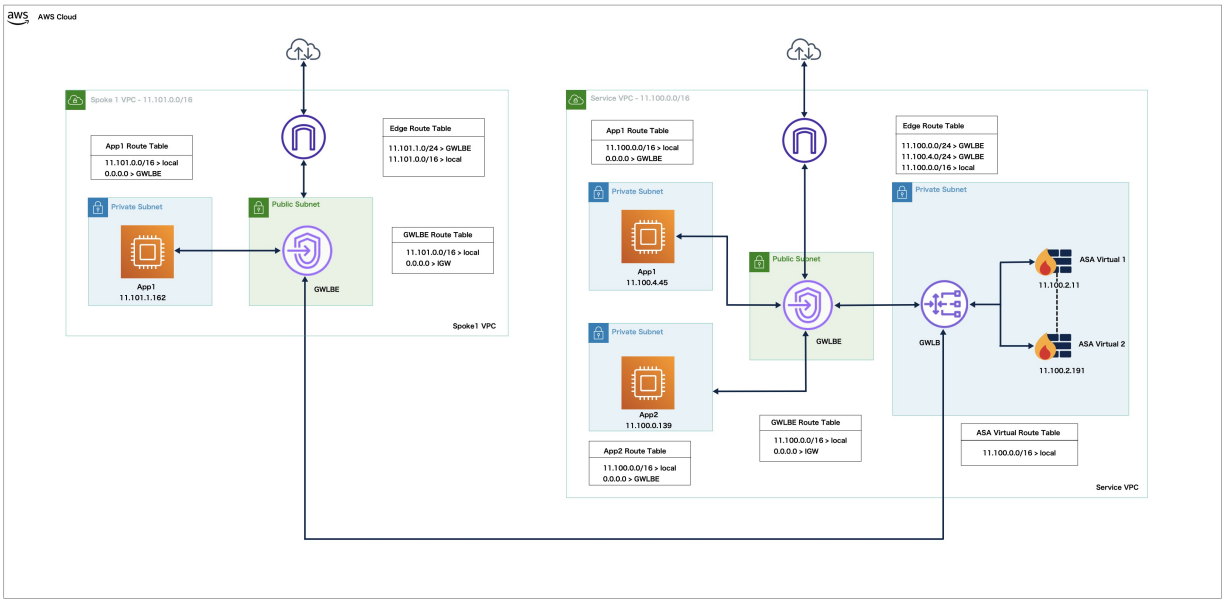
可用性区域是指一个 AWS 区域内独立运行的独立数据中心或一组独立数据中心。每个区域都有自己的网络基础设施、连接和电源，从而确保一个区域的故障不会影响到其他区域。为了提高冗余和可靠性，公司在灾难恢复计划中使用多个可用性区域。

跨多个可用性区域部署 ASA Virtual 并配置支持动态扩展的集群可以显着增强基础设施的可用性和可扩展性。此外，在同一区域利用多个可用性区域可以提供额外的冗余，并在发生故障时保证高可用性。

您可以修改集群控制链路 (CCL) 的 IP 分配机制，以支持 AWS 上 ASA Virtual 群集的单可用性区域和多可用性区域部署。下面给出的拓扑结构描述了具有自动扩展能力的单个和多个可用性区域中的入站和出站流量。

### ASA Virtual 在单个可用性区域中使用自动扩展进行集群

集群中有两个连接到 GWLB 的 ASA Virtual 实例。

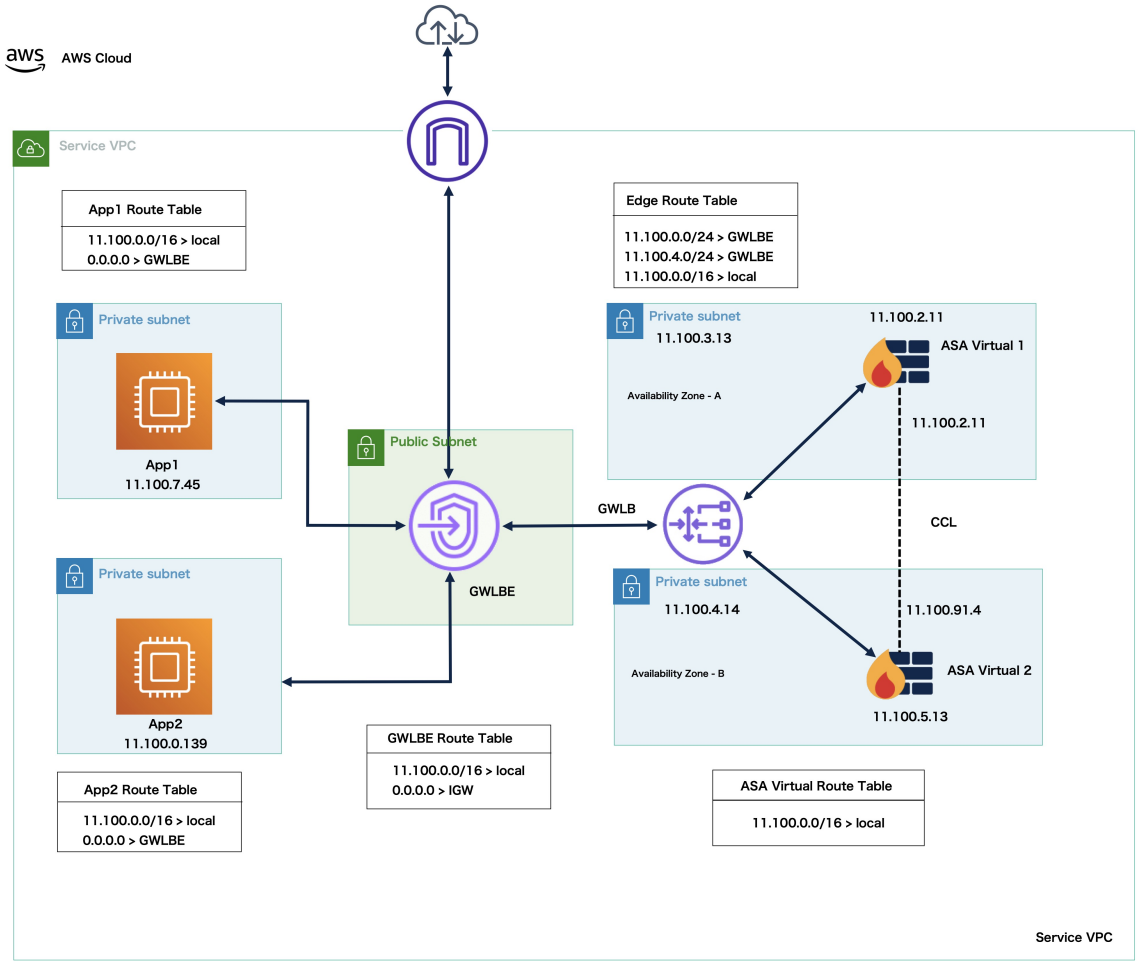



来自互联网的入站流量会进入GWLB端点，然后由端点将流量传输到GWLB。然后，流量被转发到 ASA Virtual 集群。集群中的 ASA Virtual 实例检测到流量后，将其转发到应用虚拟机 App1。

来自 App1 的出站流量将传输到 GWLB 终端 > GWLB > ASAv > GWLB > GWLB 终端，然后由 GWLB 终端发送到互联网。

**ASA Virtual 使用自动缩放解决方案在多个可用性区域中使用自动缩放进行集群**

集群中有两个 ASA Virtual 实例连接到 GWLB，它们位于不同的可用性区域。

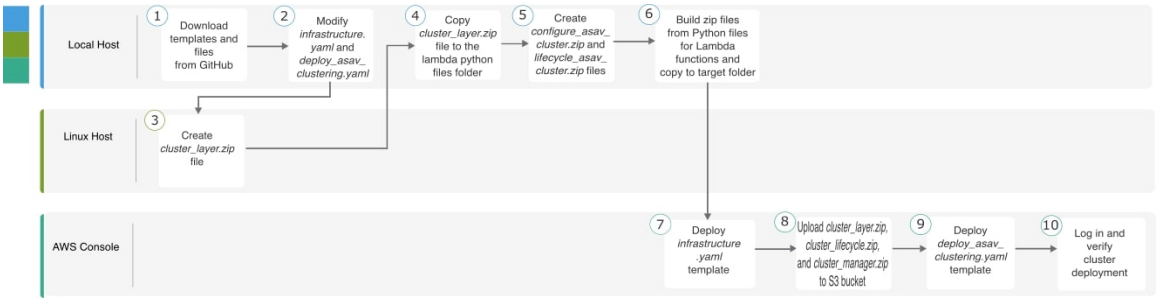



来自互联网的入站流量会进入GWLB端点，然后由端点将流量传输到GWLB。然后，根据可用性区域将流量路由到 ASA Virtual 集群。集群中的 ASA Virtual 实例检测到流量后，将其转发到应用虚拟机 App1。

# 在 AWS 上部署 ASA Virtual 集群的端到端流程

### 基于模板的部署

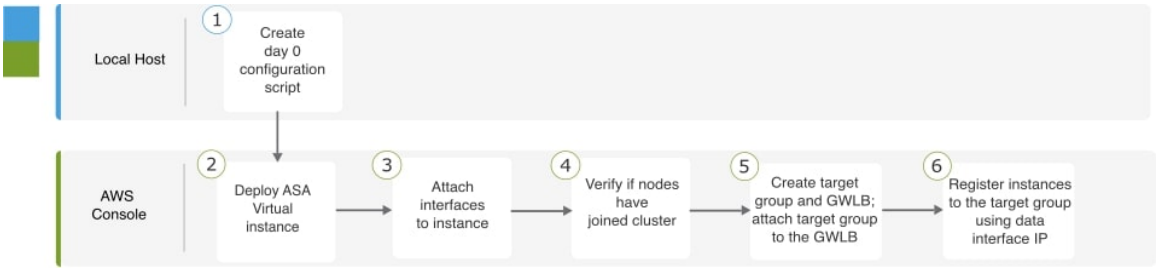
以下流程图说明了在 AWS 上基于模板部署 ASA Virtual 集群的工作流程。




| | 工作空间 | 步骤 |
|---|---|---|
| 1 | 本地主机 | 从 GitHub 下载模板和文件。 |
| 2 | 本地主机 | 修改 *infrastructure.yaml* 和 *deploying_asav_clustering.yaml* 模板。 |
| 3 | Linux 主机 | 创建 *cluster_layer.zip* 文件。 |
| 4 | 本地主机 | 将 *cluster_layer.zip* 文件复制到 Lambda python files 文件夹。 |
| 5 | 本地主机 | 创建 *configure_asav_cluster.zip* 和 *lifecycle_asav_cluster.zip* 文件。 |
| 6 | 本地主机 | 从 Python 文件为 Lambda 函数构建 zip 文件，并复制到目标文件夹。 |
| 7 | AWS 控制台 | 部署 *Infrastructure.yaml* 模板。 |
| 8 | AWS 控制台 | 将 cluster_layer.zip、cluster_lifecycle.zip 和 cluster_manager.zip 上传 S3 存储桶。 |
| 9 | AWS 控制台 | 部署 *deploy_asav_clustering.yaml* 模板。 |
| 10 | AWS 控制台 | 登录并验证集群部署。 |

### 手动部署

以下流程图说明了在 AWS 上手动部署 ASA Virtual 集群的工作流程。



| | 工作空间 | 步骤 |
|---|---|---|
| 1 | 本地主机 | 创建 Day 0 配置脚本。 |
| 2 | AWS 控制台 | 部署 ASA Virtual 实例。 |
| 3 | AWS 控制台 | 将接口连接到实例。 |
| 4 | AWS 控制台 | 验证节点是否已加入集群。 |
| 5 | AWS 控制台 | 创建目标组和 GWLB；将目标组附加到 GWLB。 |
| 6 | AWS 控制台 | 使用数据接口 IP 向目标组注册实例。 |

# 模板

以下提供的模板可在GitHub中获取。参数值一目了然，包括模板中给出的参数名称、默认值、允许值和说明。

- Infrastructure.yaml - 基础设施部署模板。
- deploy_asav_clustering.yaml - 用于集群部署的模板。

注释 在部署集群节点之前，请确保检查支持的AWS实例类型列表。此列表可在 *deploy_asav_clustering.yaml* 模板中的参数 InstanceType 的允许值下找到。

# 在 AWS 中使用 GWLB 配置 ASA Virtual 集群的目标故障转移

AWS 中的ASA 虚拟 集群利用网关负载均衡器 (GWLB) 来均衡网络数据包并将其转发到指定的 ASA 虚拟 节点。GWLB用于在目标节点发生故障切换或取消注册的情况下，继续向该节点发送网络数据包。

AWS 中的目标故障转移功能使 GWLB 能够在计划维护期间取消注册或目标节点发生故障的情况下将网络数据包重定向到正常运行的目标节点。它利用集群的状态故障切换。

注释 如果目标节点在 GWLB 使用 SSH、SCP、CURL 或其他协议路由流量时发生故障，则将流量重定向到正常运行的目标可能会出现延迟。此延迟是由于流量的重新平衡和重新路由导致的。

在 AWS 中，您可以通过 AWS ELB API 或 AWS 控制台来配置目标故障转移。

- AWS API - 在 AWS Elastic Load Balancing (ELB) API - *modify-target-group-attributes* 中，您可以通过修改以下两个新参数来定义流量处理行为。
  - target_failover.on_unhealthy - 定义当目标变得运行不正常时 GWLB 如何处理网络流量。
  - target_failover.on_deregistration - 定义当目标取消注册时 GWLB 如何处理网络流量。

  以下命令显示了定义这两个参数的 API 参数配置示例。

```
aws elbv2 modify-target-group-attributes \
--target-group-arn arn:aws:elasticloadbalancing:…/my-targets/73e2d6bc24d8a067 \
--attributes \
Key=target_failover.on_unhealthy, Value=rebalance[no_rebalance] \
Key=target_failover.on_deregistration, Value=rebalance[no_rebalance]
```

  有关详细信息，请参阅 AWS 文档中的 TargetGroupAttribute。

- AWS 控制台 – 在 EC2 控制台中，您可以通过配置以下选项来启用目标组页面上的目标故障转移选项。
  - 编辑目标组属性
  - 启用目标故障转移
  - 验证再平衡流量

  有关如何启用目标故障转移的详细信息，请参阅启用目标故障转移功能。

# 使用 CloudFormation 模板在 AWS 中部署堆栈

使用自定义 CloudFormation 模板在 AWS 中部署堆栈。

### 开始之前

- 您需要一台安装了 Python 3 的 Linux 计算机。

### 过程

**步骤 1** 准备模板。

a) 将 github 存储库克隆到本地文件夹。请参阅https://github.com/CiscoDevNet/cisco-asav/tree/master/cluster/aws。

b) 使用所需的参数修改 **infrastructure.yaml** 和 **deploy_asav_clustering.yaml**。

c) 创建名为 **cluster_layer.zip** 的文件，为 Lambda 函数提供必要的 Python 库。

我们建议使用安装了 Python 3.9 的 Amazon Linux 创建 **cluster_layer.zip** 文件。

**注释**
如果您需要 Amazon Linux 环境，可以使用 Amazon Linux 2023 AMI 或运行最新版本的 Amazon Linux 的 AWS Cloudshell 创建 EC2 实例。

要创建 cluster-layer.zip 文件，您需要先创建包含 python 库软件包详细信息的 **requirements.txt** 文件，然后运行 shell 脚本。

1. 通过指定 python 软件包详细信息来创建 **requirements.txt** 文件。

   以下是您在 **requirements.txt** 文件中提供的示例软件包详细信息：

   ```
   $ cat requirements.txt
   pycryptodome
   paramiko
   requests
   scp
   jsonschema
   cffi
   zipp
   importlib-metadata
   ```

2. 运行以下 shell 脚本以创建 **cluster_layer.zip**文件。

   ```
   $ pip3 install --platform manylinux2014_x86_64
   --target=./python/lib/python3.9/site-packages
   --implementation cp --python-version 3.9 --only-binary=:all:
   --upgrade -r requirements.txt
   $ zip -r cluster_layer.zip ./python
   ```

   注释
   如果在安装期间遇到依赖项冲突错误（例如 urllib3 或加密），建议您在 **requirements.txt** 文件中包含冲突软件包及其建议版本。之后，您可以再次运行安装来解决冲突。

d) 将生成的 **cluster_layer.zip** 文件复制到 lambda python files 文件夹。

e) 创建 **configure_asav_cluster.zip** 和 **lifecycle_asav_cluster.zip** 文件

   可以在克隆存储库顶级目录中找到 make.py 文件。这样会将 python 文件压缩为 Zip 文件并复制到目标文件夹。

   **python3 make.py build**

   注释
   在部署集群之前，请确保已根据基础设施要求（例如 ASAv 集群自动扩展部署所需的 VPC、子网、S3 存储桶）自定义 AWS CFT 模板文件 - **Infrastructure.yaml** 和 **deploy_asav_clustering.yaml**。

**步骤 2** 部署 **Infrastructure.yaml** 并记下集群部署的输出值。在部署基础设施堆栈之前，必须确定将使用的 AWS 区域和可用性区域。每个 AWS 区域都有一组不同的可用性区域和 VPC 基础设施，因此必须为部署选择正确的区域和可用性区域。

a) 在 AWS 控制台上，转到 **CloudFormation** 并点击**创建堆栈 (Create stack)**；选择**使用新资源（标准）(With new resources [standard])**。

b) 选择**上传模板文件 (Upload a template file)**，点击**选择文件 (Choose file)**，然后从目标文件夹中选择 **infrastructure.yaml**。

c) 点击**下一步 (Next)** 并提供所需的信息。

d) 为集群输入唯一的**集群名称**和**集群编号**。

e) 从**可用性区域 (Availability Zone)** 列表中选择可用性区域。此字段仅列出基于 AWS 区域的可用性区域，您选择该区域是为了使用 ClusterFormation 模板部署基础设施堆栈。

f) 在**内部子网 CIDR (Inside subnet CIDR)** 字段中输入用逗号分隔的预定义子网 CIDR 值。

g) 从 **从 Lambda AZ 列表 (List of Lambda AZs)** 中选择 Lambda 跨可用性区域通信所需的可用性区域。

h) 点击**下一步 (Next)**，然后点击**创建堆栈 (Create stack)**。

系统将显示 CREATE_COMPLETE 信息，说明基础设施堆栈的部署状态。

i) 在部署完成后，转到**输出 (Outputs)** 并记下 S3 **BucketName**。

**图 *2: infrastructure.yaml* 的输出**

**Outputs** (13)

Search outputs

| Key | Value | Description | Export name |
|---|---|---|---|
| AZ | sa-east-1a | Availability zone | - |
| BucketName | ran-cls-infra-s3bucketcluster-kckr7518u00l | Name of the Amazon S3 bucket | - |
| BucketUrl | http://ran-cls-infra-s3bucketcluster-kckr7518u00l.s3-website-sa-east-1.amazonaws.com | URL of S3 Bucket Static Website | - |
| CCLSubnetId | subnet-050feb347e57eba99 | CCL subnet ID | - |
| EIPforNATgw | 52.67.246.95 | EIP reserved for NAT GW | - |
| InInterfaceSGId | sg-0333e92f36b2aa0bf | Security Group ID for Instances Inside Interface | - |
| InsideSubnetIds | subnet-047c0a2beffb5a70f | Inside subnet ID | - |
| InstanceSGId | sg-0c0c6bfb5ba5f1c10 | Security Group ID for Instances Management Interface | - |
| LambdaSecurityGroupId | sg-01771b0d3012a40c5 | Security Group ID for Lambda Functions | - |
| LambdaSubnetIds | subnet-0fb24785c687d50e4,subnet-0f1996a02ffaa2e62 | List of lambda subnet IDs (comma seperated) | - |
| MgmtSubnetIds | subnet-02d4a757b95a9a5b9 | Mangement subnet ID | - |
| UseGWLB | Yes | Use Gateway Load Balancer | - |
| VpcName | vpc-003b592ad2518d03d | Name of the VPC created | - |

**步骤 3** 将 **cluster_layer.zip**、**configure_asav_cluster.zip** 和 **lifecycle_asav_cluster.zip** 上传到通过 **infrastructure.yaml** 创建的 S3 存储桶。

**步骤 4** 部署 **deploy_asav_clustering.yaml**：

a) 转到 **CloudFormation**，点击**创建堆栈 (Create stack)**，然后选择**使用新资源（标准）(With new resources [standard])**。

b) 依次点击**上传模板文件 (Upload a template file)**、**选择文件 (Choose file)**，然后从目标文件夹中选择 **deploy_asav_clustering.yaml**。

c) 点击**下一步 (Next)** 并提供所需的信息。

d) 提供以下集群和基础设施配置信息。

| 参数 | 允许的值/类型 | 说明 |
|---|---|---|
| 集群配置 | | |

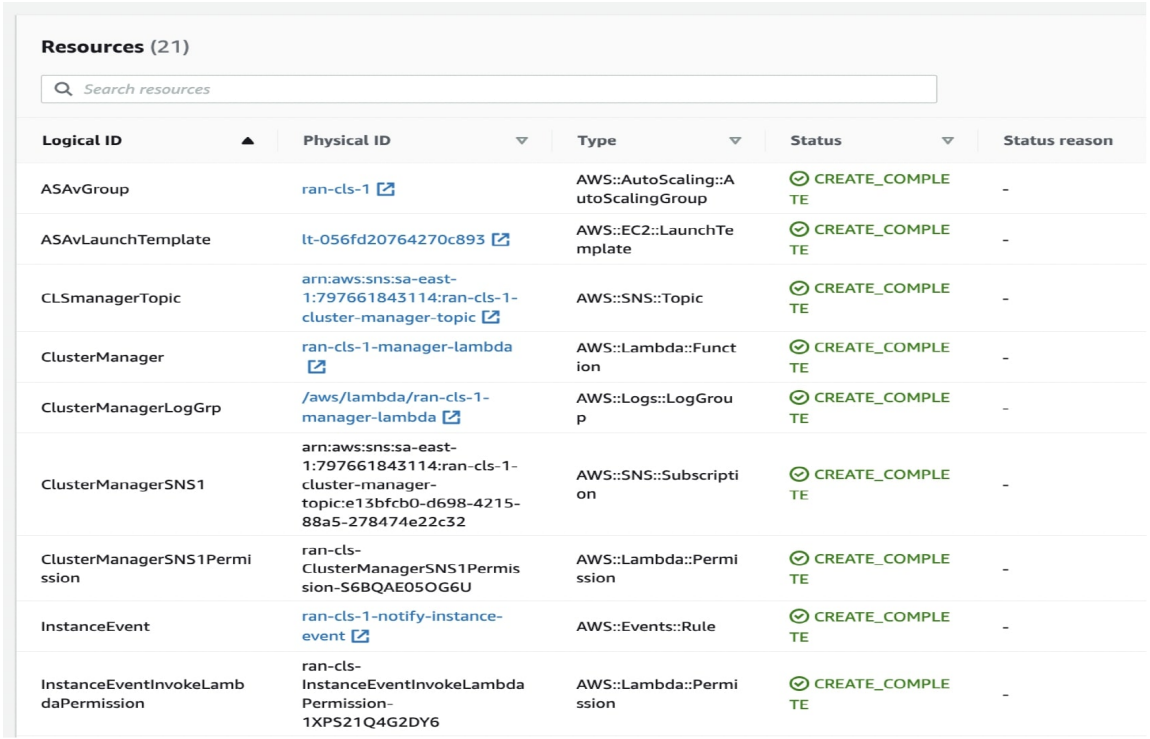
| 参数 | 允许的值/类型 | 说明 |
|---|---|---|
| 集群名称 | 字符串 | 这是集群名称前缀。集群编号将作为后缀添加。 |
| 集群编号 | 字符串 | 这是集群编号。这将作为集群名称的后缀 (msa-asa-infra)。例如，如果此值为 **1**，则组名称将为 *msa-asa-infra-1*。<br>它应至少为 1 个数字，但不超过 3 个数字。默认值：1。 |
| 集群规模 | 数字 | 这是集群中 ASA 虚拟节点的总数。<br>最小值：1<br>最大值：16 |
| **基础设施详细信息** | | |
| 可用性区域数量 | 字符串 | 这是部署ASA虚拟的可用性区域总数。（根据地区的不同，可用性区域的数量从最少 1 个到最多 3 个不等）。<br>子网将在这些可用性区域内创建。<br>此列表中可用的可用性区域基于为部署集群而选择的区域。<br>**注释**<br>Lambda 跨两个可用性区域创建。管理子网和内部子网根据此参数在三个可用性区域内创建。 |
| 输入有效的可用性区域列表 | 字符串 | 可用性区域列表基于您计划部署的区域。<br>在可用性区域列表中，选择有效的可用性区域（1 个或 2 个可用性区域或 3 个可用性区域）。<br>计数应与可用性区域数量参数的值匹配。 |
| 集群电子邮件通知 | 字符串 | 集群事件电子邮件将被发送到的邮件地址。您必须接受订用电子邮件请求，才能收到此电子邮件通知。<br>示例：admin@company.com |
| **GWLB 配置** | | |
| 部署 GWLB 终端 | 字符串 | 点击**是 (Yes)** 部署 GWLB 端点。<br>默认情况下，该值设置为**否 (No)**。 |
| GWLBE 的 VPC ID | 字符串 | 输入 VPC 以部署网关负载均衡器终端。<br>**注释**<br>如果不部署 GWLB 终端，请不要在此字段中输入任何值。 |

| 参数 | 允许的值/类型 | 说明 |
|---|---|---|
| GWLBE 的子网 ID | 字符串 | 仅输入一个子网 ID。<br>注释<br>如果不部署 GWLB 终端，请不要在此字段中输入任何值。<br>确保子网属于正确的 VPC 以及您指定的可用性区域。 |
| 目标故障转移 | 字符串 | 当目标发生故障或取消注册时，启用目标故障转移支持。（默认情况下，此参数的值设置为 **rebalance**）。<br>• **no_rebalance**：将现有流量导向故障目标，将新流量导向健康目标，从而确保向后兼容性。<br>• **rebalance**：重新分配现有流量，同时确保新流量流向正常运行的目标。 |
| 输入 GWLB 的运行状况检查端口 | 字符串 | 输入 GWLB 的运行状况检查端口。<br>注释<br>默认情况下，此端口不得用于流量。<br>确保您提供的值是有效的 TCP 端口。默认值：80 |
| **思科 ASAv 实例配置** | | |
| ASAv 实例类型 | 字符串 | 思科 ASAv EC2 实例类型。<br>确保 AWS 区域支持您选择的实例类型。<br>默认情况下，**c5.xlarge** 处于选中状态。 |
| ASAv 实例许可证类型 | 字符串 | 选择思科 ASAv EC2 实例许可证类型。确保您在 **AMI-ID** 参数中输入的 AMI ID 的许可类型相同。<br>默认情况下，选择 **BYOL**。 |
| ASAv 智能软件许可证令牌 | 字符串 | （可选）提供用于注册 ASAv 设备的思科智能软件许可证令牌。 |
| ASAv AMI-ID | 字符串 | 根据地区、版本和许可证类型（BYOL 或 PAYG）选择正确的 AMI ID。<br>ASAv 9.19 及更高版本支持集群，ASAv 9.22 及更高版本支持自动扩展和多可用性区域增强功能。<br>类型：AWS::EC2::Image::Id 默认情况下：ami-024f546cb9cbae1bb |
| ASAv 密码 | 字符串 | ASAv 实例密码。 |

| 参数 | 允许的值/类型 | 说明 |
|---|---|---|
| | | 密码将在 ASAv 可访问后激活。<br>最小长度必须为 8 个字符。密码可以是纯文本密码，也可以是 KMS 加密密码。 |
| Cpu 阈值 | CommaDelimitedList | （可选）指定非零的下阈值和上阈值将创建比例策略。如果选择 (0,0)，则不会创建 CPU 扩展警报或策略。评估点和数据点采用默认值或建议值。<br>默认情况下，此模板中启用**自动扩展**。部署后可以禁用自动扩展。 |

e) 点击**下一步**。
f) 点击以确认所有 AWS CloudFormation 选项。
g) 点击**提交 (Submit)** 以部署集群。
h) 点击**下一步 (Next)**，然后点击**创建堆栈 (Create Stack)**。

图 *3:* 已部署的资源



**Resources** (21)

Search resources

| Logical ID | Physical ID | Type | Status | Status reason |
|---|---|---|---|---|
| ASAvGroup | ran-cls-1 | AWS::AutoScaling::AutoScalingGroup | CREATE_COMPLETE | - |
| ASAvLaunchTemplate | lt-056fd20764270c893 | AWS::EC2::LaunchTemplate | CREATE_COMPLETE | - |
| CLSmanagerTopic | arn:aws:sns:sa-east-1:797661843114:ran-cls-1-cluster-manager-topic | AWS::SNS::Topic | CREATE_COMPLETE | - |
| ClusterManager | ran-cls-1-manager-lambda | AWS::Lambda::Function | CREATE_COMPLETE | - |
| ClusterManagerLogGrp | /aws/lambda/ran-cls-1-manager-lambda | AWS::Logs::LogGroup | CREATE_COMPLETE | - |
| ClusterManagerSNS1 | arn:aws:sns:sa-east-1:797661843114:ran-cls-1-cluster-manager-topic:e13bfcb0-d698-4215-88a5-278474e22c32 | AWS::SNS::Subscription | CREATE_COMPLETE | - |
| ClusterManagerSNS1Permission | ran-cls-ClusterManagerSNS1Permission-S6BQAE05OG6U | AWS::Lambda::Permission | CREATE_COMPLETE | - |
| InstanceEvent | ran-cls-1-notify-instance-event | AWS::Events::Rule | CREATE_COMPLETE | - |
| InstanceEventInvokeLambdaPermission | ran-cls-InstanceEventInvokeLambdaPermission-1XPS21Q4G2DY6 | AWS::Lambda::Permission | CREATE_COMPLETE | - |

状态从 CREATE_IN_PROGRESS 变为 CREATE COMPLETE，表示部署成功。

**步骤 5** 通过登录到任何一个节点并输入 **show cluster info** 命令来验证集群部署。

```
show cluster info
```

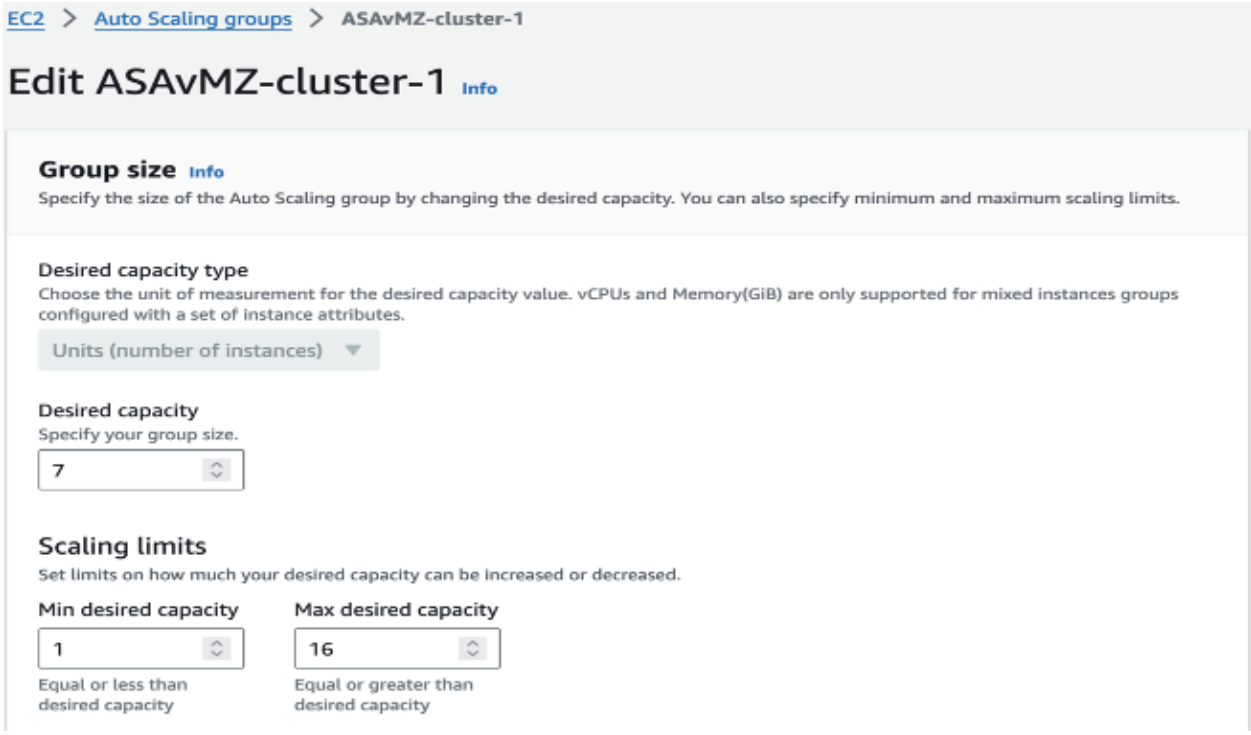
```
Cluster oneclicktest-cluster: On
Interface mode: individual
Cluster Member Limit : 16
This is "200" in state CONTROL_NODE
ID : 0
Version : 9.19.1
Serial No.: 9AU42EN5D1E
CCL IP : 1.1.1.200
CCL MAC : 4201.0a0a.0fc7
Module : ASAv
Resource : 4 cores / 8192 MB RAM
Last join : 15:26:22 UTC Jul 17 2022
Last leave: N/A
Other members in the cluster:
Unit "204" in state DATA_NODE
ID : 1
Version : 9.19.1
Serial No.: 9AJ9N46947R
CCL IP : 1.1.1.204
CCL MAC : 4201.0a0a.0fcb
Module : ASAv
Resource : 4 cores / 8192 MB RAM
Last join : 16:57:42 UTC Jul 17 2022
Last leave: 16:03:25 UTC Jul 17 2022
```

## 自动扩展参数配置

部署完成后，必须指定 ASAv Autoscale 组的**最小 (Minimum)**、**最大 (Maximum)** 和**期望 (Desired)** 容量。您必须验证自动扩展功能。

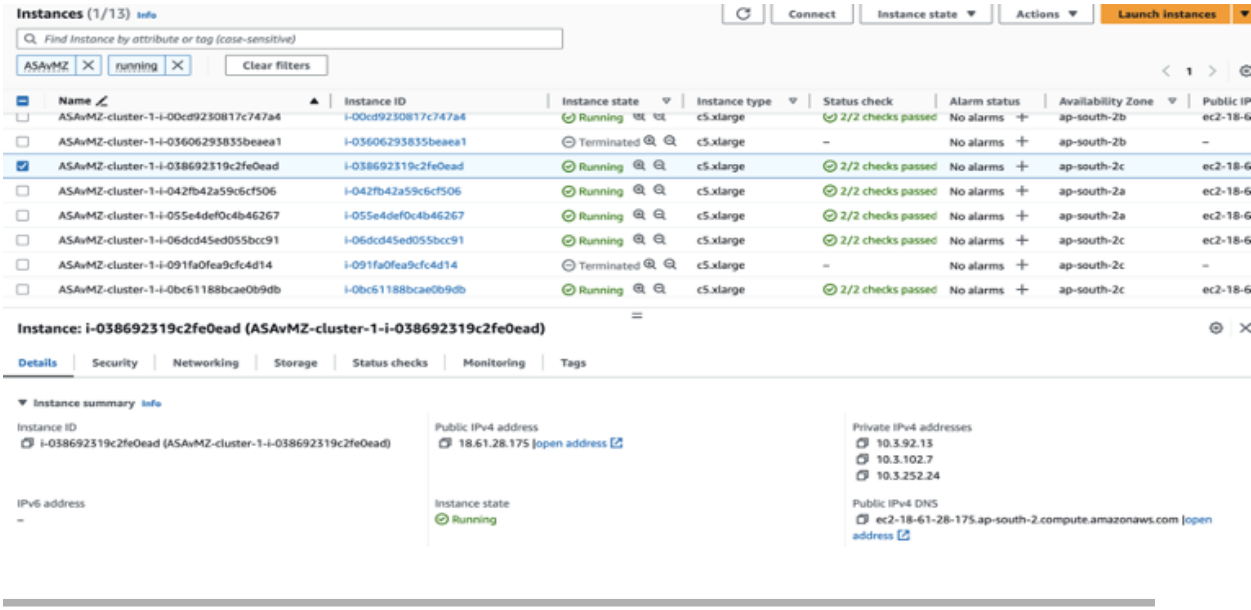
### 过程

**步骤 1** 在 AWS 控制台中，依次选择 **服务 (Services)** > **EC2** > **自动扩展组 (Auto Scaling groups)** > **已创建 ClusterAutoscale 组 (Created ClusterAutoscale group)**。




**步骤 2** 配置所需容量 **(Desired capacity)**，然后设置扩展限制 **(Scaling limits)** 容量。

**步骤 3** 在 AWS Cloudwatch 警报中检查 CPU 指标数据是否可用，缩放是否按预期进行。




## 通过更新堆叠在 **ASA Virtual** 集群中配置所需的 **IMDSv2** 模式

您可以为 AWS 上已部署的 ASA Virtual 自动缩放组实例配置 IMDSv2 必需模式。

**Before you begin**

仅 ASA Virtual 9.20.3 及更高版本支持 IMDSv2 必需模式。在为您的部署配置 IMDSv2 Required 模式之前，请确保您现有实例的版本与 IMDSv2 API 兼容（升级到 9.20.3 版或更高版本）。

**Procedure**

**步骤 1** 在 AWS 控制台中，转到 **CloudFormation**，然后点击**堆栈 (Stacks)**。

**步骤 2** 选择初始部署的集群实例的堆栈。

**步骤 3** 点击**更新**。

**步骤 4** 在**更新堆栈 (Update stack)** 页面上，点击**替换现有模板 (Replace existing template)**。

**步骤 5** 在 **指定模板** 部分下，点击 **上传模板文件**。

**步骤 6** 选择并上传支持 IMDSv2 的模板。

**步骤 7** 为模板中的输入参数提供值。

**步骤 8** 更新堆栈。

# 在 AWS 中手动部署集群

要手动部署集群，请准备 Day-0 配置并部署每个节点。

## 创建 AWS 的 Day-0 配置

使用以下命令为每个集群节点提供引导程序配置：

### 网关负载均衡器示例

以下运行配置示例会为网关负载均衡器创建一个配置，其中一个用于掉头流量的Geneve接口和一个用于集群控制链路的 VXLAN 接口。

```
cluster interface-mode individual force
policy-map global_policy
class inspection_default
no inspect h323 h225
no inspect h323 ras
no inspect rtsp
no inspect skinny

int m0/0
management-only
nameif management
security-level 100
ip address dhcp setroute
no shut

interface TenGigabitEthernet0/0
nameif geneve-vtep-ifc
security-level 0
ip address dhcp
```

```
no shutdown

interface TenGigabitEthernet0/1
nve-only cluster
nameif ccl_link
security-level 0
ip address dhcp
no shutdown

interface vni1
description Clustering Interface
segment-id 1
vtep-nve 1

interface vni2
proxy single-arm
nameif ge
security-level 0
vtep-nve 2

object network ccl_link
range 10.1.90.4 10.1.90.254 //Mandatory user input, use same range on all nodes
object-group network cluster_group
network-object object ccl_link
nve 2
encapsulation geneve
source-interface geneve-vtep-ifc
nve 1
encapsulation vxlan
source-interface ccl_link
peer-group cluster_group

cluster group asav-cluster // Mandatory user input, use same cluster name on all nodes
local-unit 1 //Value in bold here must be unique to each node
cluster-interface vni1 ip 1.1.1.1 255.255.255.0 //Value in bold here must be unique to each
 node
priority 1
enable noconfirm

mtu geneve-vtep-ifc 1806
mtu ccl_link 1960
aaa authentication listener http geneve-vtep-ifc port 7575 //Use same port number on all
nodes
jumbo-frame reservation
wr mem
```

注释 对于 AWS 运行状况检查设置，请务必指定您在此处设置的 **aaa authentication listener http** 端口。

### 非本地负载均衡器示例

以下示例会创建一个配置，用于具有管理接口、内部接口和外部接口的非本地负载均衡器，以及用于集群控制链路的 VXLAN 接口。

```
cluster interface-mode individual force
interface Management0/0
management-only
nameif management
ip address dhcp
```

```
interface GigabitEthernet0/0
no shutdown
nameif outside
ip address dhcp

interface GigabitEthernet0/1
no shutdown
nameif inside
ip address dhcp

interface GigabitEthernet0/2
nve-only cluster
nameif ccl_link
ip address dhcp
no shutdown

interface vni1
description Clustering Interface
segment-id 1
vtep-nve 1

jumbo-frame reservation
mtu ccl_link 1654
object network ccl_link
range 10.1.90.4 10.1.90.254          //mandatory user input
object-group network cluster_group
network-object object ccl_link

nve 1
encapsulation vxlan
source-interface ccl_link
peer-group cluster_group

cluster group asav-cluster          //mandatory user input
local-unit 1                        //mandatory user input
cluster-interface vni1 ip 10.1.1.1 255.255.255.0  //mandatory user input
priority 1
enable
```

注释 如果要复制并粘贴上面给出的配置，请确保从配置中删除 **//mandatory user input**。

## 部署集群节点

部署集群节点以便形成集群。

### 过程

**步骤 1** 使用具有所需数量的接口的集群 Day 0 配置部署 ASA Virtual 实例 - 如果使用网关负载均衡器 (GWLB)，则为三个接口；如果使用非本地负载均衡器，则为四个接口。在 **配置实例详细信息 > 高级详细信息** 部分中，粘贴您的 day0 配置。

**注释**
确保按以下顺序将接口连接到实例。

- AWS 网关负载均衡器 - 三个接口 - 管理、诊断、内部和集群控制链路。

- 非本地负载均衡器 - 四个接口 - 管理、内部、外部和集群控制链路。

有关在 AWS 上部署 ASA Virtual 的更多信息，请参阅 在 AWS 上部署 ASA Virtual。

**步骤 2** 重复步骤 1 以部署所需数量的其他节点。

**步骤 3** 使用 ASA Virtual 控制台上的 **show cluster info** 命令验证是否所有节点都已成功加入集群。

**步骤 4** 配置 AWS 网关负载均衡器。

a) 创建目标组和 GWLB。

b) 将目标组连接到 GWLB。

**注释**
确保将 GWLB 配置为使用正确的安全组、侦听程序配置和运行状况检查设置。

c) 使用IP地址向目标组注册数据接口（内部接口）。有关详细信息，请参阅创建网关负载均衡器。

## 在 AWS 中为 ASA Virtual 启用目标故障转移

ASA Virtual 的数据接口已注册到 AWS 中的 GWLB 目标组。在 ASA Virtual 集群中，每个实例都与一个目标组关联。GWLB 进行负载平衡，并将流量发送到该运行状况正常的实例，而该实例已被识别或注册为目标组中目标节点。

### 开始之前

您必须已通过手动方法或使用 CloudFormation 模板在 AWS 中部署 ASA Virtual 堆栈。

如果您使用 CloudFormation 模板部署集群，您还可以通过分配集群部署文件 `deploy_asav_clustering.yaml` 的 **GWLB 配置**部分下提供的 **rebalance** 属性来启用 **Target Failover** 参数。在模板中，默认情况下，此参数的值设置为 **rebalance**。但在 AWS 控制台上，此参数的默认值设置为 **no_rebalance**。

其中，

- **no_rebalance** - GWLB 继续将网络流量发送到发生故障或已取消注册的目标。
- **重新平衡**- 当现有目标发生故障或取消注册时，GWLB 将网络流量发送到另一个正常运行的目标。

有关在 AWS 中部署堆栈的信息，请参阅：

- 在 AWS 中手动部署集群
- 使用 CloudFormation 模板在 AWS 中部署堆栈

### 过程

**步骤 1** 在 AWS 控制台上，转到**服务 (Services) > EC2**

**步骤 2** 点击**目标组 (Target Groups)** 以查看目标组页面。

**步骤 3** 选择并打开 ASA Virtual 实例 IP 地址注册到的目标组。系统将显示目标组详细信息页面。

**步骤 4** 转到**属性 (Attributes)** 菜单。

**步骤 5** 点击**编辑 (Edit)** 以编辑属性。

**步骤 6** 将**再平衡流 (Rebalance flows)** 滑块按钮切换到右侧。这使目标故障切换能够配置 GWLB，以便在目标故障切换或注销注册时，将现有网络数据包重新平衡并转发到正常运行的目标节点。

# 在 Azure 中部署集群

在 Azure 服务链中，ASA Virtual 充当可以拦截互联网和客户服务之间的数据包的透明网关。Azure 上的 ASA Virtual 实例集群可通过将多节点 ASAv 抽象化为单个设备，从而帮助它们扩展吞吐量。

ASAv 包含两个逻辑接口 - 面向互联网的**外部接口**和面向客户服务的**内部接口**。这些接口可通过使用成对代理中的 VXLAN 网段在 ASAv 的单个网络接口卡 (NIC) 上定义。

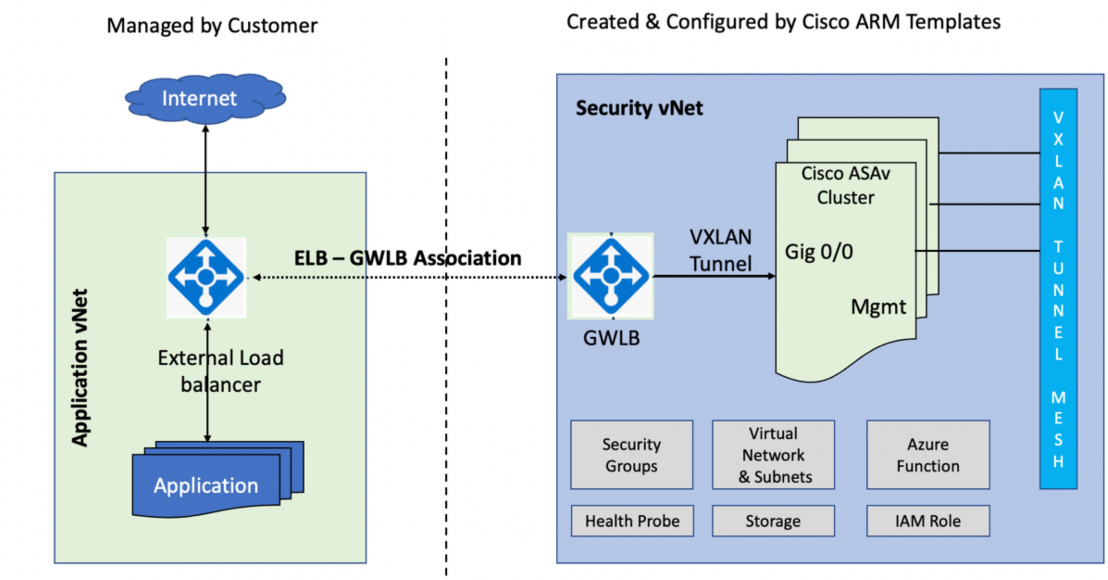
### 关于 Azure 网关负载均衡器

通过将 VXLAN 网段路由到 ASAv 进行流量检查，Azure 网关负载均衡器 (GWLB) 可帮助您均衡和管理入站和出站流量。在 ASAv 集群环境中，Azure GWLB 会根据流量负载自动扩展 ASAv 节点的吞吐量级别。GWLB 可以确保流量对称或路由到网络虚拟设备，而无需手动更新路由。使用此功能，数据包可以在两个方向上遍历同一网络路径。

下图显示了从外部 VXLAN 网段上的公共网关负载均衡器转发到 Azure GWLB 的流量。**网关负载均衡器**主要在多个 ASAv 之间平衡流量，这些 ASAv 会在丢弃流量或将其送回内部 VXLAN 网段的 GWLB 之前对流量进行检查。然后，Azure GWLB 会将流量发送回公共网关负载均衡器和目的地。

下图说明了 Azure 中 GWLB 和 ASAv 之间的网络流量。

图 *4:* 具有 *GWLB* 的 *Azure* 上的 *ASAv* 集群




# 关于 Azure 中的集群部署

您可以使用自定义的 Azure 资源管理器 (ARM) 模板为 Azure GWLB 部署虚拟机规模集。

在集群部署后，您可以使用 day0 配置或通过 Azure 门户上的函数应用来手动配置集群上的每个节点。

# 使用 Azure 资源管理器模板来部署集群

部署集群节点（虚拟机规模集），使它们使用 Azure 资源管理器 (ARM) 模板来形成集群。

### 开始之前

- 要手动创建 Azure 集群，您必须准备具有 day0 配置的配置文本文件。请参阅为 Azure 上的集群部署准备配置文件。

### 过程

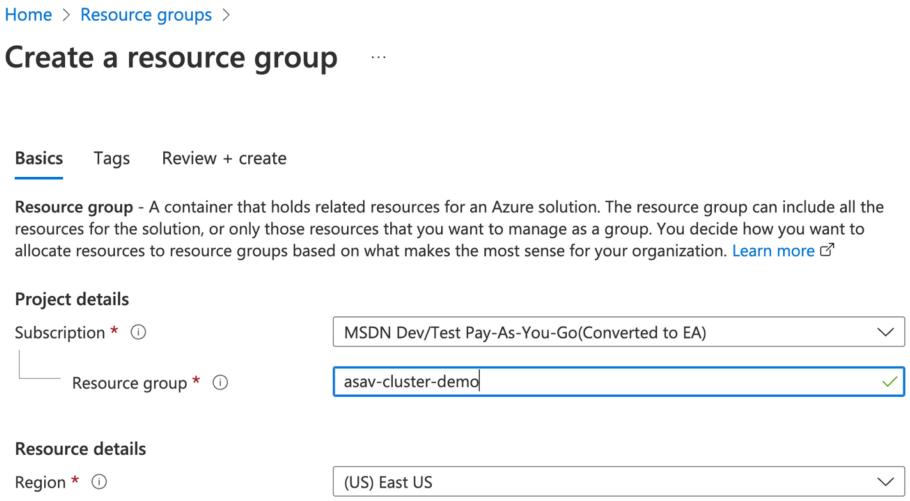
**步骤 1** 准备模板。

a) 将 GitHub 存储库克隆到本地文件夹。请参阅https://github.com/CiscoDevNet/cisco-asav/tree/master/cluster/azure。

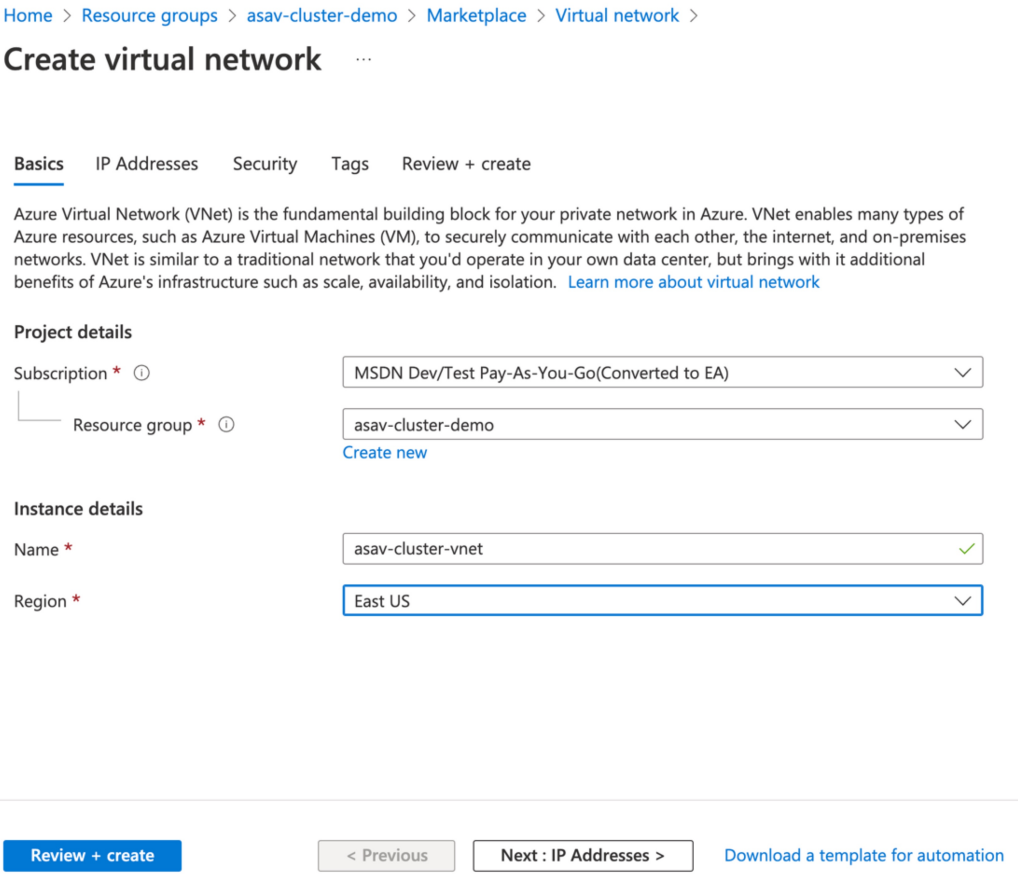
b) 对于 GWLB，请使用所需参数来修改 `azure_asav_gwlb_cluster.json` 和 `asav-gwlb-cluster-config.txt`。

**步骤 2** 登录 Azure 门户：https://portal.azure.com。

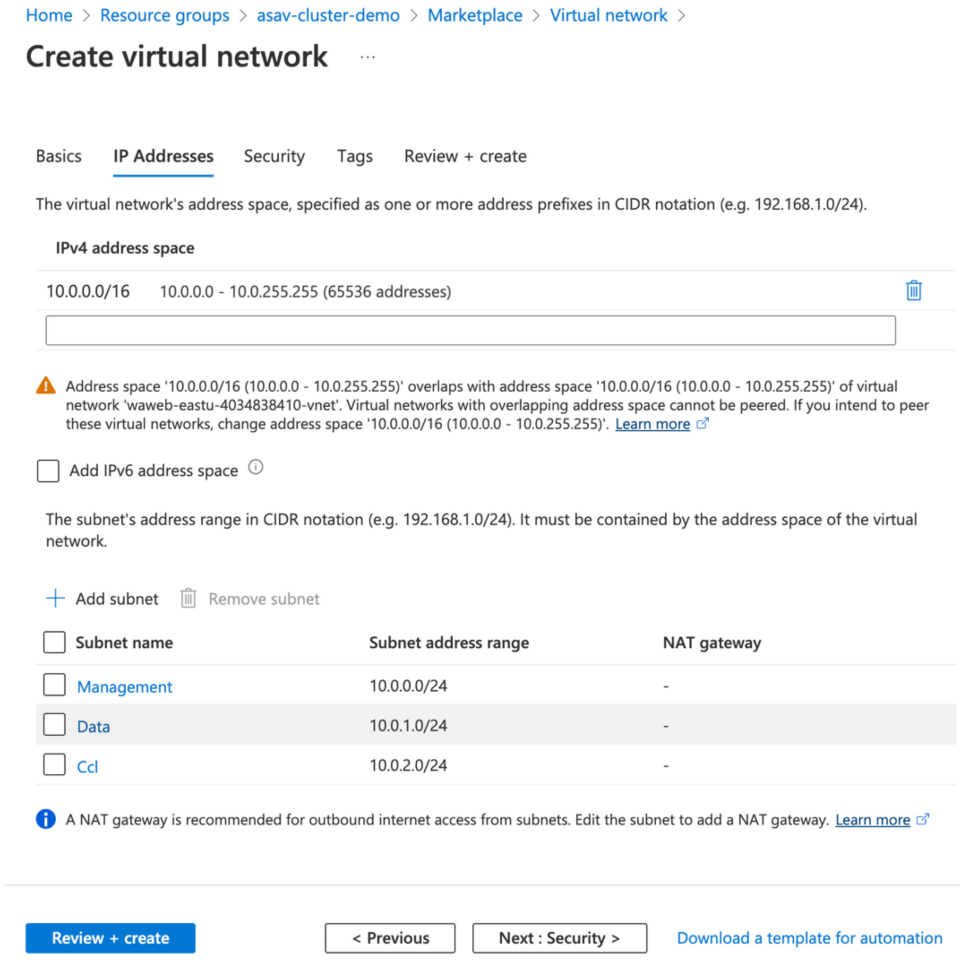
**步骤 3** 创建一个资源组。




**步骤 4** 为 ASAv 集群创建具有三个子网的虚拟网络：管理、外部和集群控制链路 (CCL)。




**步骤 5** 添加子网。




**步骤 6** 部署自定义模板。

a) 点击**创建 (Create) > > 模板部署 (Template deployment)**（使用自定义模板部署）。

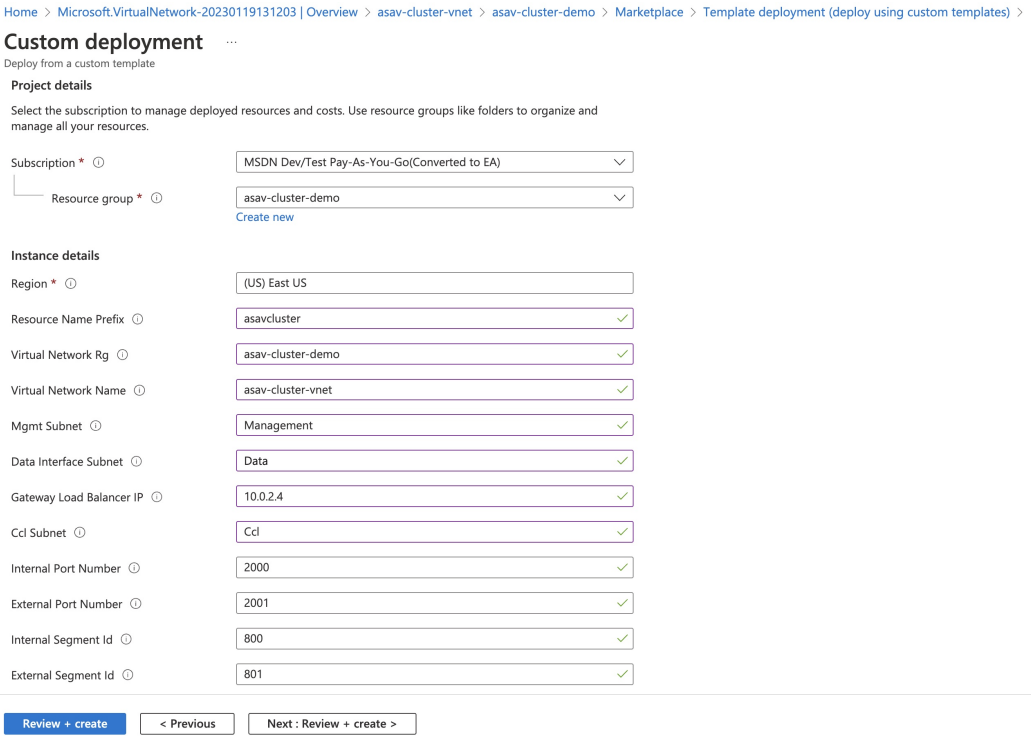
b) 点击“在编辑器中生成自己的模板 (Build your own template in the editor)”。

c) 点击**加载文件 (Load File)**，然后根据 Azure 中使用的负载均衡器类型上传 `azure_asav_gwlb_cluster.json`。

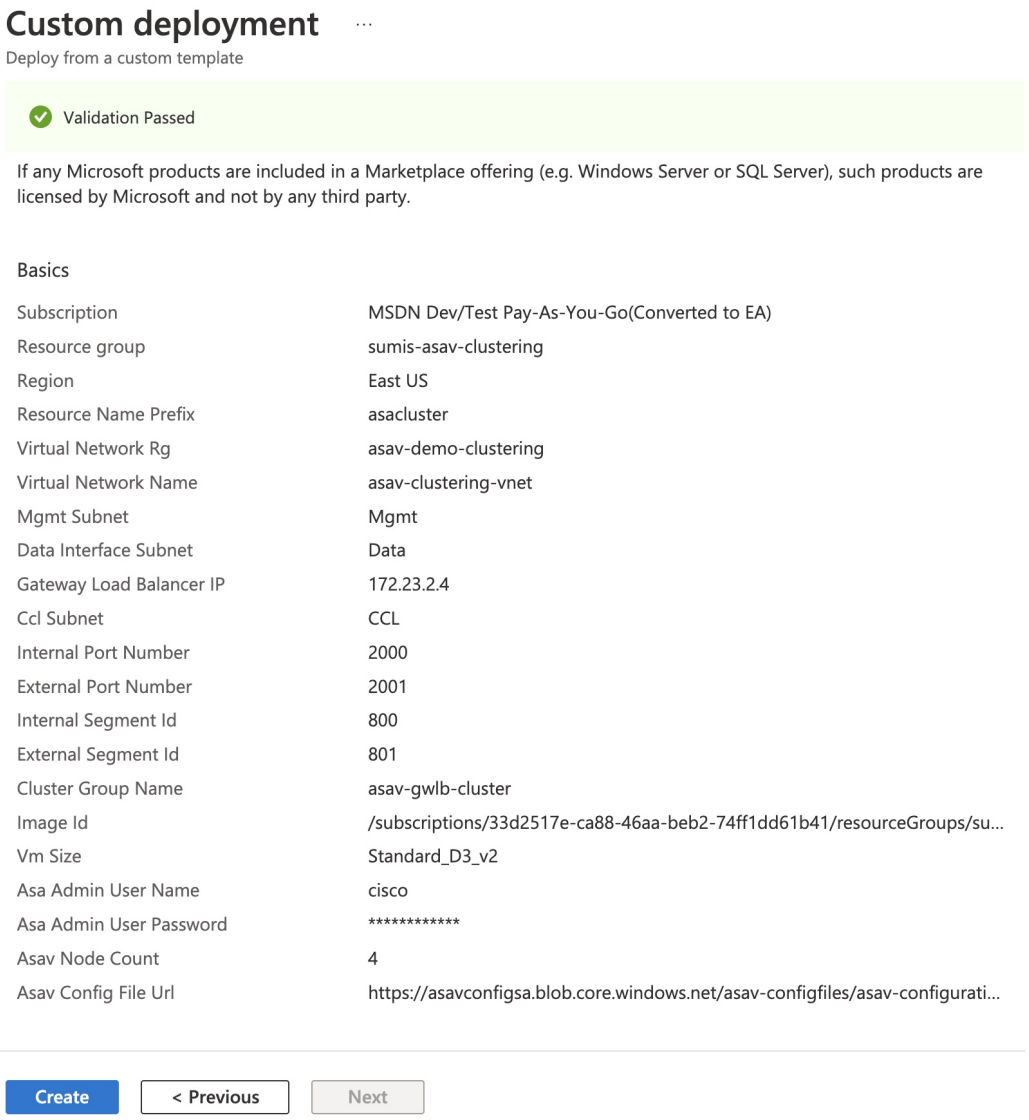
d) 点击**保存 (Save)**。

**步骤 7** 配置实例详细信息。

**步骤 8** 输入所需的值，然后点击**查看 + 创建 (Review + create)**。




**步骤 9** 在验证通过后，点击**创建 (Create)**。




**步骤 10** 在实例运行后，通过登录到任何一个节点并输入 **show cluster info** 命令来验证集群部署。

```
> show cluster info
Cluster gwlb-cluster-template-with-AN: On
    Interface mode: individual
Cluster Member Limit : 16
    This is "12" in state CONTROL_NODE
        ID        : 0
        Version   : 99.19(1)180
        Serial No.: 9AKGFV8VH4G
        CCL IP    : 10.1.1.12
        CCL MAC   : 000d.3a55.5470
        Module    : NGFWv
        Resource  : 8 cores / 28160 MB RAM
        Last join : 11:13:24 UTC Sep 5 2022
        Last leave: N/A
```

### 下一步做什么

在 Azure 中配置集群，第 32 页。

# 在 Azure 中配置集群

要在 Azure 中的 ASAv 节点上配置集群，您可以使用配置文件或使用 Azure 函数应用来进行手动配置。您可以将集群与本地 GWLB 配合使用。

## 准备用于在 Azure 上创建集群的配置文件

您可以使用配置文件或 Azure 门户上的函数应用在 ASA Virtual 节点上手动配置集群。

要在 ASA Virtual 节点上手动配置集群，您必须已配置 `asav-gwlb-cluster-config.txt` 。在此文件中，您必须定义在集群 ASA Virtual 节点中配置的参数，例如范围对象、day0、集群组名称、许可类型等。

本节介绍了如何创建集群配置文件，以使用 GWLB 配置 Azure 中的 ASA Virtual 节点。

### 过程

**步骤 1** 从思科 GitHub 存储库目录 `asav-cluster/sample-config-file` 下载 **asav-gwlb-cluster-config.txt** 。

**步骤 2** 您可以为创建集群准备 day0 配置。

以下的 day0 配置示例可帮助您了解使用 GWLB 在 Azure 中创建集群所需的参数。

- **用于 GWLB 集群创建的 Day0 配置示例**

  以下是用于 GWLB 集群创建的 `asav-gwlb-cluster-config.txt` 文件中所需的 day0 配置示例。

```
cluster interface-mode individual force
            policy-map global_policy
            class inspection_default
            no inspect h323 h225
            no inspect h323 ras
            no inspect rtsp
            no inspect skinny

        interface GigabitEthernet0/0
            nameif vxlan_tunnel
            security-level 0
            ip address dhcp
            no shutdown

        interface GigabitEthernet0/1
            nve-only cluster
            nameif ccl_link
            security-level 0
            ip address dhcp
            no shutdown

        interface vni1
```

```
    description ClusterInterface
    segment-id 1
    vtep-nve 1

interface vni2
    proxy paired
    nameif GWLB-backend-pool
    internal-segment-id 800
    external-segment-id 801
    internal-port 2000
    external-port 2001
    security-level 0
    vtep-nve 2

object network ccl#link
    range <CCLSubnetStartAddress> <CClSubnetEndAddress>
    object-group network cluster#group
    network-object object ccl#link

nve 1
    encapsulation vxlan
    source-interface ccl_link
    peer-group cluster#group

nve 2
    encapsulation vxlan
    source-interface vxlan_tunnel
    peer ip <GatewayLoadbalancerIp>

mtu vxlan_tunnel 1454
mtu ccl_link 1374
cluster group <ClusterGroupName>
local-unit <Last Octet of CCL Interface IP>
cluster-interface vni1 ip 1.1.1.<Last Octet of CCL Interface IP> 255.255.255.0

priority 1
enable
```

在上述 day0 配置示例中，当提及封装类型为 **vxlan** 时，则 GWLB 相关配置已启用。**InternalPort** 和 **ExternalPort** 用于 vxlan 隧道接口配置，而 **InternalSegId** 和 **ExternalSegId** 用作内部和外部接口的标识符。

**注释**

在 day0 配置中，必须指定集群控制链路的起始地址 (`<CCLSubnetStartAddress>`) 和结束地址。因此，`StartAddress` 必须始终以 `x.x.x.4` 开头，并且 `EndAddress` 必须在最佳范围内。建议仅指定所需数量的地址（最多 16 个），因为添加大量地址可能会影响性能。

例如：如果 CCL 子网是 `192.168.3.0/24`，则 `StartAddress` 将为 `192.168.3.4` 而 `EndAddress` 会是 `192.168.3.30`。

以下是 vni 接口所需的配置示例。

```
interface vni2
        proxy paired
        nameif GWLB-backend-pool
        internal-segment-id 800
        external-segment-id 801
        internal-port 2000
        external-port 2001
        security-level 0
```

```
vtep-nve 2
```

**步骤 3** 将配置文件上传到 Azure 存储，并记下该位置的路径 (URL)。在 ASA Virtual 节点上手动配置集群时需要此 URL 路径。

## 使用配置文件手动配置集群

使用配置文件在 Azure 中手动配置 ASAv 节点上的集群。

### 开始之前

您必须已准备好配置文件，并记录下上传文件的 Azure 存储位置。请参阅“为 Azure 准备集群配置文件”。

### 过程

**步骤 1** 登录到 Azure 门户。

**步骤 2** 打开 Azure 上部署的 ASAv 实例。

**步骤 3** 运行以下命令，通过提供已上传到 Azure 存储容器的文件的 URL 将集群配置文件复制到 ASAv 节点。

**copy <Config File URL> running-config**

**步骤 4** 运行以下命令以便在 ASAv 实例上配置集群

```
cluster group <ClusterGroupName>
         local-unit <Last Octet of the Management Interface IP>
         cluster-interface vni1 ip 1.1.1.<Last Octet of the Management Interface IP>
255.255.255.0
         priority 1
         enable
```

**步骤 5** 重复步骤 2 至 4，在所有 ASAv 节点上配置集群。

## 使用 Azure 函数应用来配置集群

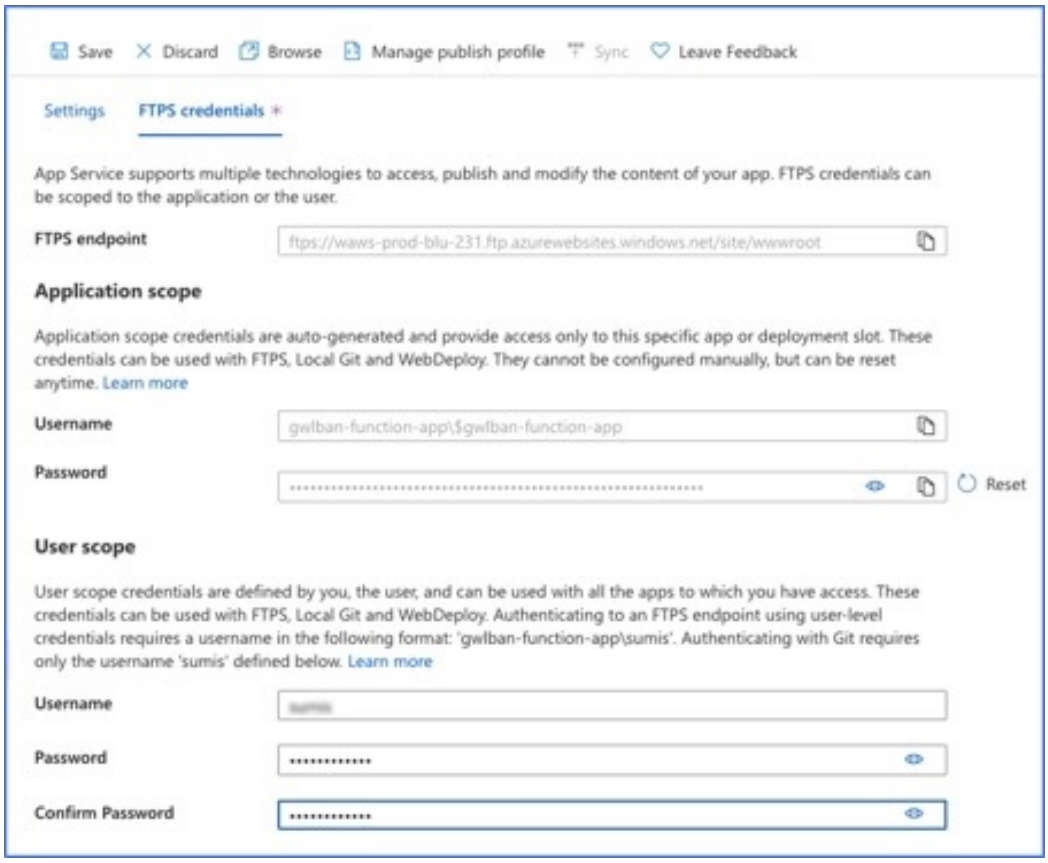
使用 Azure 函数应用服务在 Azure 中的 ASAv 节点上配置集群。

### 过程

**步骤 1** 登录到 Azure 门户。

**步骤 2** 点击**函数应用 (Function App)**。

**步骤 3** 通过点击**部署中心 (Deployment Center)** > **FTPS 凭证 (FTPS credentials)** > **用户范围 (User scope)** > **配置用户名和密码 (Configure Username and Password)** > 来创建 FTPS 凭证，然后点击**保存 (Save)**。




步骤 4 通过在本地终端中执行以下命令，将 `Cluster_Function.zip` 文件上传到函数应用。

**curl -X POST -u <Userscope_Username> --data-binary @"Cluster_Function.zip" https://<Function_App_Name>.scm.azurewebsites.net/api/zipdeploy**

图 *5:* 功能

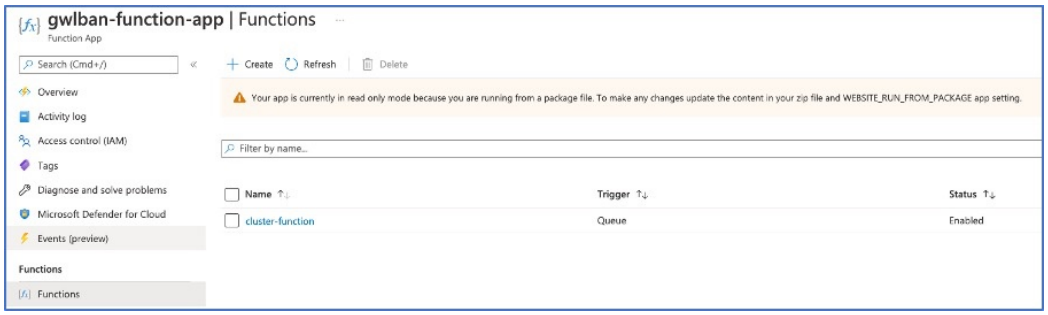


图 *6:* 队列

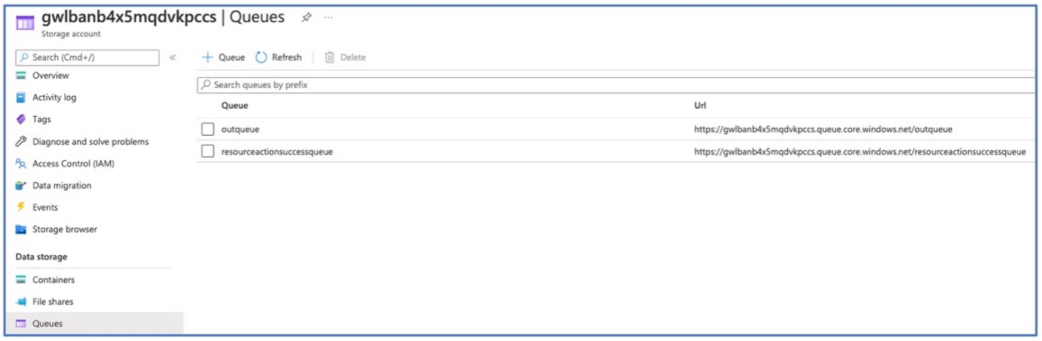
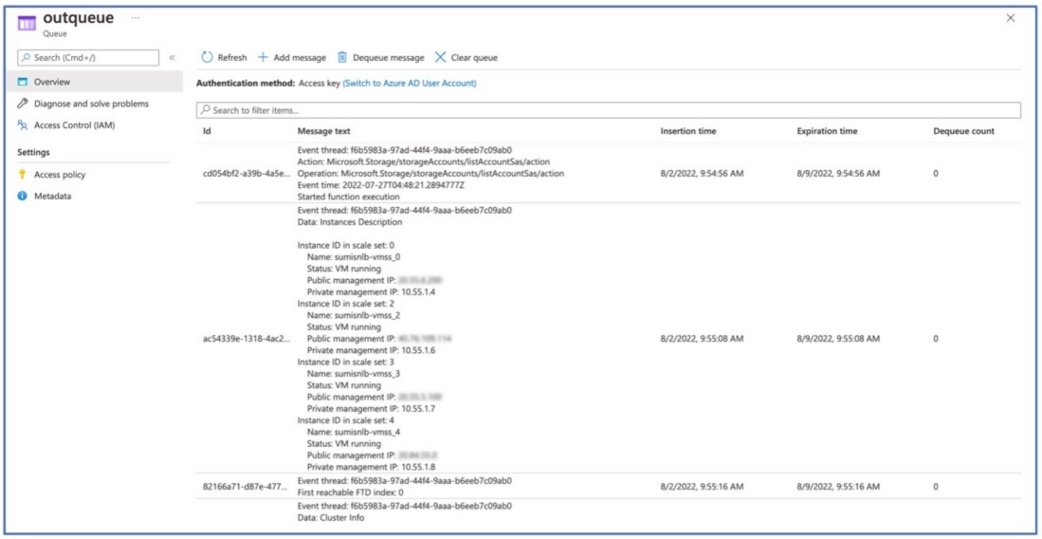



图 *7:* 出站队列




函数将被上传到函数应用。功能将启动，而您可以在存储帐户的出站队列中看到日志。

在执行此功能后，将在所有 ASAv 节点上启用集群。

# Azure 中的 ASA Virtual 集群故障排除

### 流量问题

如果流量不起作用，请验证以下内容：

1. 使用负载均衡器验证 ASA Virtual 实例的运行状况探测状态是否正常。

   如果 ASA Virtual 实例的运行状况探测状态为不正常，请执行以下操作：

   1. 验证 ASA Virtual 中配置的静态路由。
   2. 验证默认网关是否为数据子网的网关 IP。

3. 确保 ASA Virtual 实例接收运行状况探测流量。
4. 验证 ASA Virtual 中配置的访问策略是否允许运行状况探测流量。

### 集群问题

如果未形成集群，请验证以下内容：

- 网络虚拟化终端（仅 NVE）集群接口的 IP 地址。确保您可以 ping 其他节点的仅 NVE 集群接口。
- 仅 NVE 集群接口的 IP 地址是对象组的一部分。确保通过对象组来配置 NVE。
- 集群组中的集群接口具有正确的 VNI 接口。此 VNI 接口具有相应对象组的 NVE。
- 每个节点都有自己的IP接口，请验证节点是否应该能够相互执行ping操作，以确保集群中节点之间的连接。
- 验证模板部署期间提及的CCL子网的开始地址和结束地址是否正确。起始地址必须以子网中的第一个可用的 IP 地址开头。例如，如果子网是 192.168.1.0/24。那么起始地址应为 192.168.1.4（前三个 IP 地址由 Azure 保留）

### 角色相关问题

如果在同一资源组中再次部署资源时出现任何与角色相关的错误，请执行以下操作：

如果存在与特定角色相关的任何问题，则会显示错误消息。

以下是错误消息的示例。

*"error": {*

*"code": "RoleAssignmentUpdateNotPermitted",*

*"message": "Tenant ID, application ID, principal ID, and scope are not allowed to be updated." }*

通过从终端执行以下命令来删除下列角色。

- 用于删除存储队列数据参与者角色的命令：

  **az role assignment delete --resource-group <Resource Group Name> --role "Storage Queue Data Contributor"**

- 用于删除参与者角色的命令：

  **az role assignment delete --resource-group <Resource Group Name> --role "Contributor"**

# ASA Virtual Azure 上的集群自动扩展解决方案

Azure 区域中的典型集群部署包括规定数量的 ASA Virtual 实例（节点）。当 Azure 区域流量变化时，如果没有动态扩展（自动扩展）节点，这种集群安排中的资源利用率可能未充分利用资源或导致延迟。思科在 9.23 及更高版本中为 ASA Virtual 集群提供自动扩展解决方案，支持动态扩展 Azure

区域中的节点。它允许您根据网络流量从集群内向内扩展或横向扩展节点。它使用基于 Azure VMSS 指标（如 CPU 和内存指标）的资源利用率统计信息的逻辑，动态添加或删除集群中的节点。

Azure 中使用自动扩展解决方案的 ASA Virtual 集群同时支持网络负载均衡器（NLB 或三明治拓扑）和网关负载均衡器 (GWLB)。请参阅拓扑示例，第 38 页

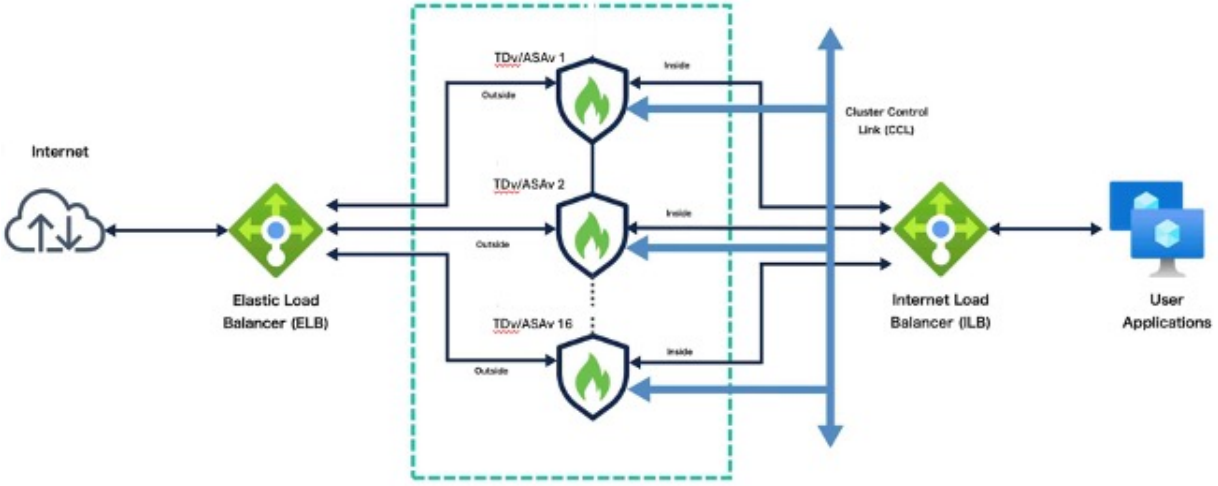
思科提供单独的 Azure 资源管理器 (ARM) 模板用于使用 NLB 和 GWLB 在 Azure 中部署具有自动扩展功能的 ASA Virtual 集群，并提供基础设施和配置模板用于部署函数应用和逻辑应用等 Azure 服务。

# 拓扑示例

### ASA Virtual 使用三明治拓扑（网络负载均衡器）在 Azure 中通过自动扩展建立集群

Azure 中使用三明治拓扑结构 (NLB) 的自动扩展的 ASA Virtual 集群用例是一种自动水平缩放解决方案，它将 ASA Virtual 规模集置于 Azure 内部负载均衡器 (ILB) 与 Azure 外部负载均衡器 (ELB) 之间。

在此拓扑中，ASA Virtual 仅使用四个接口：管理、内部、外部和 CCL 子网。




### ASA Virtual 使用三明治拓扑 (NLB) 在 Azure 中通过自动扩展建立集群

下面简要介绍了 ASA Virtual集群如何使用 NLB 功能在 Azure 中进行自动扩展：

- ELB 将流量从互联网分发到规模集中的 ASA Virtual 实例；然后，防火墙将流量转发到应用。
- ILB 将出站互联网流量从应用分发到规模集中的 ASA Virtual 实例；然后，防火墙将流量转发到互联网。
- 网络数据包决不会在一个连接中同时穿过（内部和外部）负载均衡器。
- 规模集中的 ASA Virtual实例数将根据负载条件自动进行扩展和配置。

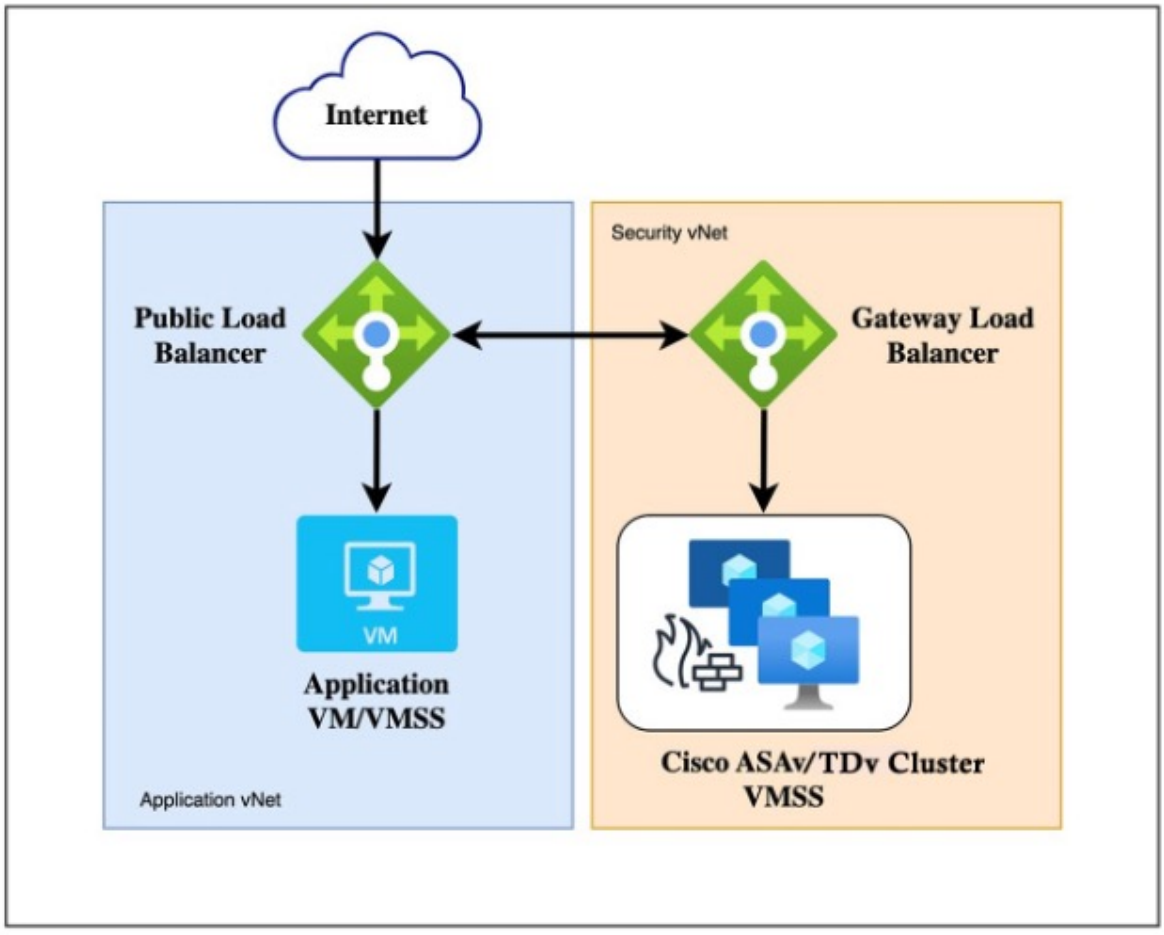
### ASA Virtual 使用网关负载均衡器在 Azure 中建立自动扩展集群

使用自动扩展解决方案的 Azure 网关负载均衡器 (GWLB) 与 ASA Virtual 集群集成可简化集群设置中的实例部署、管理和扩展。Azure 网关负载均衡器 (GWLB) 可确保 Cisco Secure Firewall 检查进出 Azure VM（例如应用服务器）的互联网流量，而无需更改任何路由。这种集成还降低了操作复杂性，并为防火墙上的流量提供了单一的入口和出口点。应用和基础设施可以保持源 IP 地址的可视性，而这在某些环境中至关重要。

ASA Virtual 在此使用案例中仅使用三个接口：管理、数据和 CCL 接口。

注释

- 如果要部署 Azure GWLB，则不需要网络地址转换 (NAT)。
- 仅支持 IPv4。




下面简要介绍了 ASA Virtual集群如何使用 GWLB 功能在 Azure 中进行自动扩展：

- 来自互联网的入站流量会进入 GWLB 端点，然后由端点将流量传输到 GWLB。

- 然后，流量将路由到ASA Virtual集群。自动扩展解决方案应用内向扩展或外向扩展逻辑，以便根据流量负载在集群中添加或删除节点。
- 集群中的 ASA Virtual 实例检测到流量后，将其转发到应用虚拟机。

# Azure 中的 ASA Virtual 集群自动扩展逻辑

### 扩展策略

在具有自动扩展的集群中，节点的扩展根据以下策略确定：

- 扩展策略 1 - 当一个集群节点超过资源使用限制时。
- 扩展策略 2 - 所有节点的总体平均资源利用率。

### 横向扩展

横向扩展是指当流量负载阈值超过任何一个集群节点上配置的CPU或内存限制时，向集群添加新节点的过程。

以下是在横向扩展期间向集群添加新节点的过程：

1. 已启动一个新的 ASA Virtual 实例。
2. 相应配置将被应用于 ASA Virtual。
3. 已应用适当的许可证。
4. 一个新的 ASA Virtual 实例将被添加到集群中。

如果在横向扩展过程中，新的ASA Virtual实例的配置失败（低概率），则会终止失败的实例，并启动和配置一个新实例。

### 内向扩展

内向扩展是指当配置的内向扩展阈值和群集实例总数超过最小集群规模时，将节点从群集中移除的过程。

以下是在内向扩展过程中终止集群中一个节点的过程：

1. 使用 VMSS 指标确定 CPU 或内存使用率最低的 ASA Virtual 实例。
2. 如果有多个实例具有相同的最低利用率，则选择 VMSS 中 VM 索引较高的实例进行内向扩展。
3. 通过适当的配置和策略，该实例的任何新连接都会被禁用。
4. 实例会从智能许可中注销（适用于 BYOL）。
5. 实例将被终止。

#### Azure Functions（函数应用）

Function 应用可帮助启用 ASA Virtual 群集并在管理中心注册。Function 应用还可以帮助您为支持自动扩展部署的 ASA Virtual 集群选择托管计划。

提供以下两种类型的托管计划：

- 使用量
  - 这是具有自动扩展功能的 ASA Virtual 集群的默认托管计划。
  - 该计划允许 Function 应用通过打开 SSH 端口连接到该区域的 Azure 数据中心 IP 地址，从而连接到 ASA Virtual 实例。
- 高级
  - 您可以在部署期间为 Function 应用选择此托管计划。
  - 此计划支持将网络地址转换 (NAT) 网关添加到 Function 应用，以控制 Function 应用的出站 IP 地址。此计划仅允许从 NAT 网关的固定 IP 地址对 ASA Virtual 实例进行 SSH 访问，从而增强安全性。

# GitHub 上的部署和基础设施模板

思科提供 Azure 资源管理器 (ARM) 模板和脚本，用于使用多个 Azure 服务（包括函数应用、逻辑应用、自动扩展组等）部署 ASA Virtual 集群的自动扩展组。

ASA Virtual 集群的自动扩展解决方案是基于 ARM 模板的部署，可向 GWLB 和 NLB 负载平衡器提供支持。

### 使用自动扩展解决方案模板的 ASA Virtual 集群

#### Azure 资源管理器 (ARM) 模板

根据您在 Azure 中为集群使用的（NLB 或 GWLB）负载均衡器，为自动扩展解决方案提供了两套模板。

GitHub 上提供了以下模板：

- 使用 NLB 的 ASA Virtual 集群的自动扩展解决方案模板：
  `azure_ftdv_nlb_cluster_autoscale.json`，可从文件夹
  `azure_autoscale_clustering/tdv_cluster/arm_templates/` 获取
- 使用 GWLB 的 ASA Virtual 集群的自动扩展解决方案模板：
  `azure_ftdv_gwlb_cluster_autoscale.json`，可从文件夹
  `azure_autoscale_clustering/tdv_cluster/arm_templates/` 获取

#### GitHub 上的 Azure 基础设施和配置模板

以下是在 Azure 上为使用自动扩展功能的集群设置 Azure 基础设施所需的模板。

- 用于在 ASA Virtual 实例上启用集群的函数应用：`cluster_functions.zip`，可从文件夹 `azure_autoscale_clustering/tdv_cluster/azure_function_app` 获取。
- 用于 ASA Virtual 部署、内向扩展和横向扩展工作流程的逻辑应用代码：`logical_app.txt`，可从文件夹 `azure_autoscale_clustering/tdv_cluster/logic_app/` 获取。

# 输入参数

下表定义了模板参数并提供了示例。确定这些值后，就可以在将 Azure 资源管理器 (ARM) 模板部署到 Azure 订阅时使用这些参数来创建 ASA Virtual 设备。在使用 GWLB for Azure 的自动扩展集群中，还创建了网络基础设施，因此必须在模板中配置额外的输入参数。参数说明的含义不言自明。

**表 *2:* 模板参数**

| 参数名 | 允许的值/类型 | 说明 | 资源创建类型 |
|---|---|---|---|
| resourceNamePrefix | 字符串*（3-10 个字符） | 所有资源都使用包含此前缀的名称创建。<br>注：只能使用小写字母。<br>示例：asav | New |
| virtualNetworkRg | 字符串 | 虚拟网络资源组名称。<br>示例：cisco-virtualnet-rg | 现有 |
| virtualNetworkName | 字符串 | 虚拟网络名称（已创建）。<br>示例：cisco-virtualnet | 现有 |
| mgmtSubnet | 字符串 | 管理子网名称（已创建）<br>示例：cisco-mgmt-subnet | 现有 |
| dataSubnet | 字符串 | 数据子网名称（已创建）<br>示例：cisco-data-subnet | |
| cclSubnet | 字符串 | 集群控制链路子网名称。<br>示例：cisco-ccl-subnet | |
| cclSubnetStartAddr | 字符串 | CCL 子网 IP 地址的起始范围。<br>示例：3.4.5.6 | |
| cclSubnetEndAddr | 字符串 | CCL 子网 IP 地址的结束范围。<br>示例：5.6.7.8 | |

| 参数名 | 允许的值/类型 | 说明 | 资源创建类型 |
|---|---|---|---|
| gwlbIP | 字符串 | GWLB 已在现有数据子网中创建。<br>示例：10.0.2.4 | |
| dataNetworkGatewayIp | 字符串 | 数据子网的网关 IP 地址。<br>示例：10.0.2.7 | |
| insideSubnet | 字符串 | 内部子网名称（已创建）。<br>示例：cisco-inside-subnet | 现有 |
| internalLbIp | 字符串 | 内部子网的内部负载均衡器 IP 地址（已创建）。<br>例如：1.2.3.4 | 现有 |
| outsideSubnet | 字符串 | 外部子网名称（已创建）。<br>示例：cisco-outside-subnet | 现有 |
| softwareVersion | 字符串 | ASA Virtual 版本（在部署期间从下拉列表中选择）。 | 现有 |
| vmSize | 字符串 | ASA Virtual 实例的大小（在部署过程中从下拉列表中选择）。 | 不适用 |
| asaAdminUserName | 字符串* | ASA Virtual 'admin' 用户的用户名。<br>这**不能**是“admin”。请参阅 Azure 以了解 VM 管理员用户名准则。<br>**注释**<br>模板中不对此进行合规性检查。 | New |
| asaAdminUserPassword | 字符串* | ASA Virtual管理员用户的密码。<br>密码的长度必须为 12 至 72 个字符，而且必须具有：小写、大写、数字及特殊字符；重复字符不得超过 2 个。<br>**注释**<br>模板中不对此进行合规性检查。 | New |

| 参数名 | 允许的值/类型 | 说明 | 资源创建类型 |
|---|---|---|---|
| clusterGroupName | 字符串 | 向管理中心注册威胁防御设备时使用的集群组名称。<br>示例：asav-cluster | |
| asaLicensingSku | 字符串 | ASA Virtual 的许可模式（PAYG 或 BYOL）。 | |
| healthCheckPortNumber | 字符串 | 在网关负载均衡器中创建运行状况探测器时使用的运行状况检查端口号。<br>示例：8080 | |
| functionHostingPlan | 字符串 | 功能部署托管计划（消耗使用使用量托管计划，高级：使用高级托管计划）。<br>默认值：功耗 | |
| functionAppSubnet | 字符串 | 函数应用子网名称（已创建）。<br>示例：asav-fapp-subnet | |
| functionAppSubnetCIDR | 字符串 | 函数应用子网的 CIDR（已创建）。<br>示例：10.0.4.0/24 | |
| scalingMetricsList | 字符串 | 用于确定扩展决定的指标。<br>允许：CPU | |
| scalingPolicy | POLICY-1/POLICY-2 | **POLICY-1**：当任何 ASA Virtual 的平均负载在所配置的持续时间内超过外向扩展阈值时，将触发外向扩展。<br>**POLICY-2**：当自动扩展组中所有 ASA Virtual 设备的平均负载在所配置的持续时间内超过外向扩展阈值时，将触发外向扩展。<br>在两种情况下，内向扩展逻辑都保持不变：当所有 ASA Virtual 设备的平均负载在所配置的持续时间内低于内向扩展阈值时，将触发内向扩展。 | 不适用 |

| 参数名 | 允许的值/类型 | 说明 | 资源创建类型 |
| --- | --- | --- | --- |
| scalingMetricsList | 字符串 | 用于制定扩展决策的指标。<br>允许：CPU<br>默认值：CPU | 不适用 |
| scaleInThreshold | 字符串 | CPU 指标的内向扩展阈值。<br>默认值：10<br>当 ASA Virtual 指标低于此值时，将触发扩展。 | 不适用 |
| scaleOutThreshold | 字符串 | CPU 指标的横向扩展阈值。<br>默认值：80<br>当ASA Virtual指标高于此值时，将触发横向扩展。<br>“scaleOutThreshold”应始终大于“scaleInThreshold”。 | 不适用 |
| asavClusterSize | 字符串 | 在任何给定时间，扩展集中可用的 ASA Virtual 实例的默认节点数。<br>示例：4 | |
| minAsaCount | 整数 | 在任何给定时间，规模集中可用的最小 ASA Virtual 实例数。<br>示例：2 | 不适用 |
| maxAsaCount | 整数 | 规模集中允许的最大ASA Virtual实例数。<br>示例：10<br>**注释**<br>Auto Scale逻辑不会检查此变量的范围，因此请认真填写。 | 不适用 |

| 参数名 | 允许的值/类型 | 说明 | 资源创建类型 |
|---|---|---|---|
| metricsAverageDuration | 整数 | 从下拉列表中选择。<br>此数字表示计算指标平均值的时间（以分钟为单位）。<br>如果此变量的值为 5（即 5 分钟），则当计划 Auto Scale Manager 时，它将检查过去 5 分钟内的指标平均值，并且基于此平均值做出扩展决定。<br>**注释**<br>由于 Azure 限制，仅 1、5、15 和 30 是有效数字。 | 不适用 |
| initDeploymentMode | BULK/STEP | 主要适用于第一次部署，或者规模集不包含任何 ASA Virtual 实例时。<br>BULK：Auto Scale 管理器将尝试一次并行部署“minAsaCount”数量的 ASA Virtual 实例。<br>STEP：Auto Scale 管理器将按照计划间隔逐个部署“minAsaCount”数量的 ASA Virtual 设备。 | |
| smartLicenseToken | 字符串 | 用于注册 ASA Virtual 的智能许可证令牌。 | |
| licenseThroughput | 字符串 | ASA Virtual 的智能许可证权利层级。 | |
| asavConfigFileUrl | 字符串 | ASA Virtual 配置文件的完整路径。<br>示例：<br>https://path_to_asav_config_file/config_file<br>确保配置文件可从 ASAv 访问。 | 不适用 |
| *Azure 对新资源的命名约定有限制。查看限制，或者直接全部使用小写字母。**不要使用空格或任何其他特殊字符**。 | | | |

# ASA Virtual 具有自动扩展部署过程和资源的集群

ASA Virtual 在 Azure 上部署具有自动扩展功能的集群涉及以下内容：

- 部署 ARM 模板。
- 构建并部署集群功能。
- 更新并启用逻辑应用。

### Azure 资源管理器模板部署资源

对于**三明治拓扑 (NLB)** - `azure_ftdv_nlb_cluster_autoscale.json`，当使用 ARM 模板在 Azure 中部署具有自动扩展功能的 ASA Virtual 集群时，会在资源组中创建以下资源

- 虚拟机规模集 (VMSS)
- 外部负载均衡器
- 内部负载均衡器
- Azure 函数应用
- 逻辑应用
- 安全组（用于数据接口和管理接口）

当使用 GWLB - `azure_ftdv_gwlb_cluster_autoscale.json`在 Azure 中部署具有自动扩展功能的 ASA Virtual 集群时，会在资源组中创建以下资源

- 虚拟机 (VM) 或虚拟机规模集 (VMSS)
- 网关负载均衡器 (GWLB)
- Azure 函数应用
- 逻辑应用
- 网络基础设施
- 部署所需的安全组和其他各种组件。

# 使用自动扩展解决方案部署 ASA Virtual 集群

使用 ARM 模板在 Azure 上部ASA Virtual署具有自动扩展解决方案的 Threat Defense Virtual 集群。根据拓扑结构、三明治 (NLB) 使用案例或 GWLB 使用案例，您需要下载并配置适当的 ARM 模板，以便在 Azure 上部署支持自动扩展的 ASA Virtual 集群解决方案。

### 开始之前

**从 GitHub 下载部署软件包**

面向 Azure 的使用 NLB 解决方案的 ASA Virtual 集群自动扩展是一个基于 Azure 资源管理器 (ARM) 模板的部署，它会利用 Azure 提供的无服务器基础设施（逻辑应用、Azure 函数、负载均衡器、虚拟机扩展设置等）。

面向 Azure 的使用 GWLB 解决方案的 ASA Virtual 集群是一种基于 ARM 模板的部署，可创建 GWLB、网络基础设施、Threat Defense Virtual 自动扩展组、无服务器组件和其他所需资源。

两种解决方案的部署过程均类似。

下载启动面向 Azure 的使用自动扩展解决方案的 ASA Virtual 集群所需的文件。

您的版本的部署脚本和模板可从 GitHub 存储库获取。https://github.com/CiscoDevNet/cisco-asav/tree/master/cluster/azure

过程

**步骤 1** 使用您的 Microsoft 帐户用户名和密码登录 Microsoft Azure 门户 (https://portal.azure.com)。

**步骤 2** 点击服务菜单中的**资源组 (Resource groups)** 以访问**资源组 (Resource Groups)** 边栏选项卡。您将看到该边栏选项卡中列出您的订用中的所有资源组。创建一个新的资源组或选择一个现有的空资源组。例如，**secure-firewall-asav-demo**。

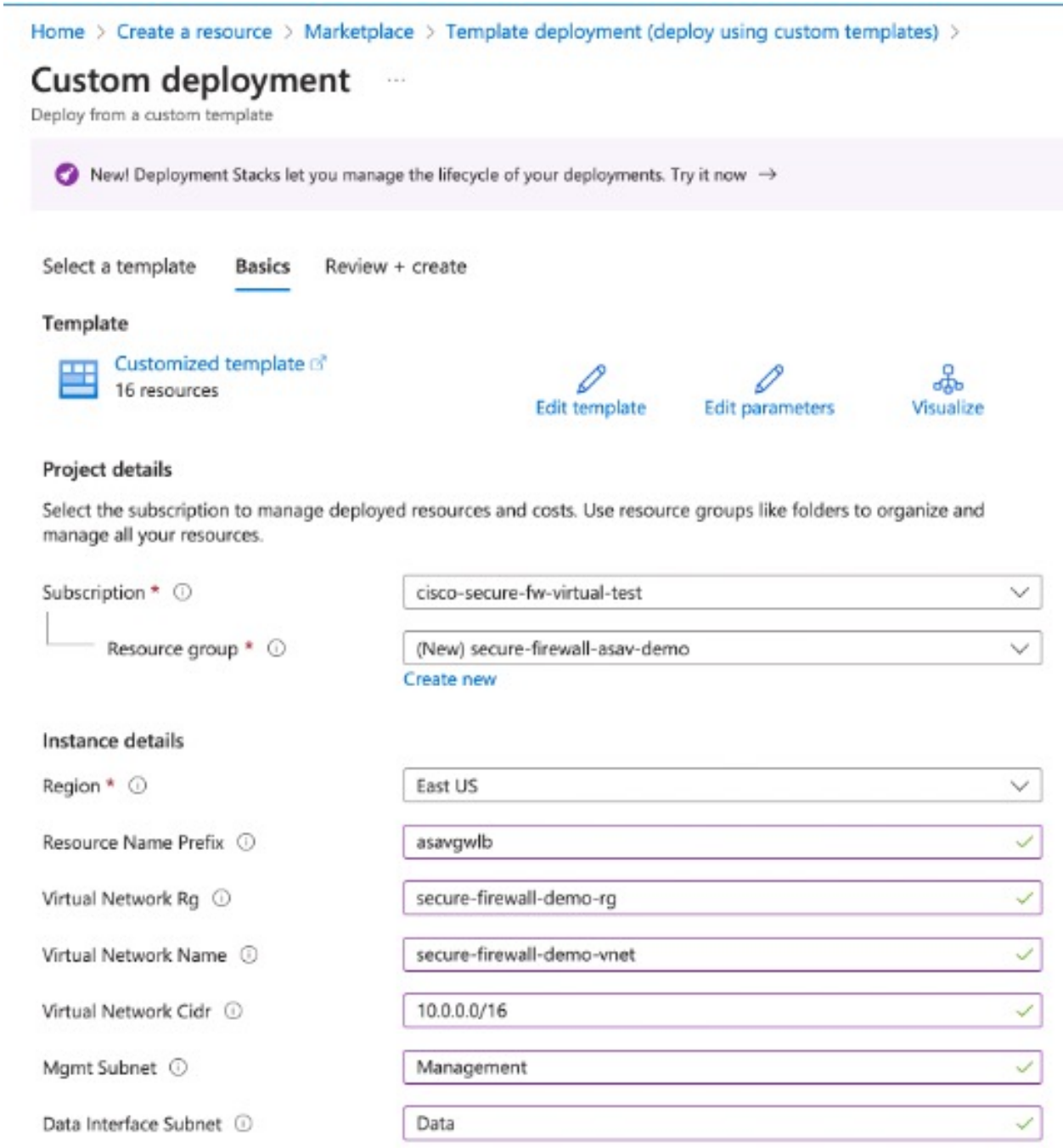
**步骤 3** 点击**创建资源 (+) (Create a resource [+])**，为模板部署创建新资源。此时将显示**创建资源组 (Create Resource Group)** 边栏选项卡。

**步骤 4** 点击服务菜单中的**虚拟网络 (Virtual Network)**，以访问“虚拟”(Virtual) 网络边栏选项卡。使用子网创建虚拟网络。

- 对于 GWLB 部署，请创建包含管理子网、数据子网和 CCL 子网的虚拟网络。
- 对于 NLB 部署，请创建包含管理、内部、外部和 CCL 子网的虚拟网络。

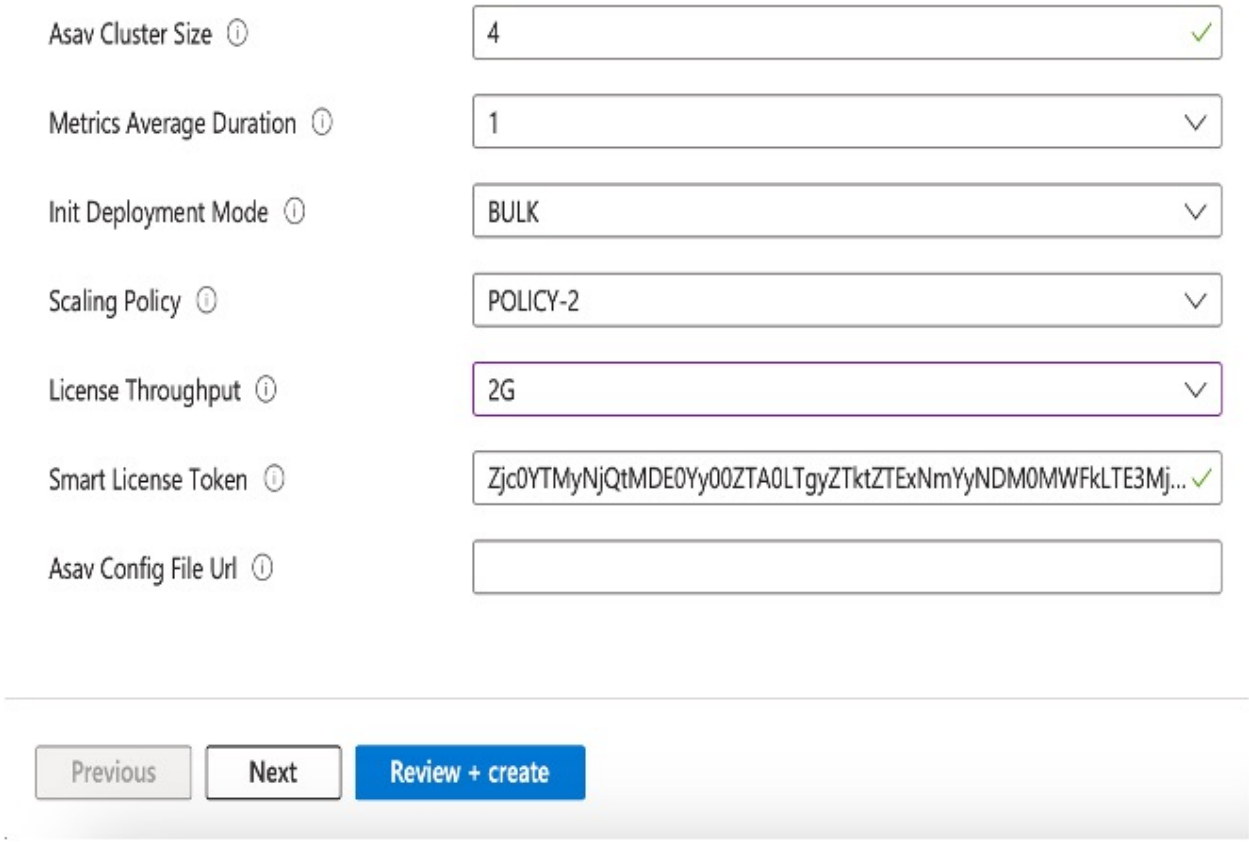
**步骤 5** 在**搜索市场 (Search the Marketplace)** 中，键入**模板部署 (`Template deployment`)**（使用自定义模板部署），然后按 **Enter**。

**步骤 6** 点击**创建 (Create)**。创建模板时有多个选项。选择**在编辑器中选择构建您自己的模板 (Build your own template in editor)**。




**步骤 7** 在**编辑模板 (Edit template)** 窗口中，删除所有默认内容并从更新的 `azure_ftdv_gwlb_cluster_custom_image.json` 或 `azure_ftdv_nlb_cluster_custom_image.json`（具体取决于您在 Azure 上部署的自动扩展解决方案的类型）复制内容，然后点击**保存 (Save)**。或者点击**加载文件 (Load file)**，从您的计算机浏览并上传此文件。

| | |
|---|---|
| Gateway Load Balancer IP | 10.0.1.4 |
| Data Network Gateway Ip | 10.0.1.1 |
| Ccl Subnet | Ccl |
| Ccl Subnet Start Addr | 10.0.3.4 |
| Ccl Subnet End Addr | 10.0.3.24 |
| Internal Port Number | 2000 |
| External Port Number | 2001 |
| Internal Segment Id | 800 |
| External Segment Id | 801 |
| Function Hosting Plan | consumption |
| Function App Subnet | |
| Function App Subnet CIDR | |
| Cluster Group Name | asavgwlb-cls |
| Image Id | /subscriptions/1fdf9165-db4d-4fc9-814b-8475c5adc637/resourceGro... |
| Vm Size | Standard_D4_v2 |
| Asa Admin User Name | cisco |
| Asa Admin User Password | •••••••••••• |
| Scale In Threshold | 10 |
| Scale Out Threshold | 60 |
| Asav Cluster Size | 4 |
| Metrics Average Duration | 1 |




**步骤 8** 在参数字段部分中填写所有参数。有关输入参数，第 42 页每个参数的详细信息，请参阅**输入参数**，然后点击查看+创建 (Review+Create)。

**步骤 9** 模板部署成功后，就会为 Azure 解决方案创建 ASA Virtual 自动扩展所需的所有资源。“类型”(Type) 列描述了每个资源，包括逻辑应用、VMSS、负载均衡器、公共 IP 地址等。

# 将 Azure 集群自动扩展函数部署到函数应用

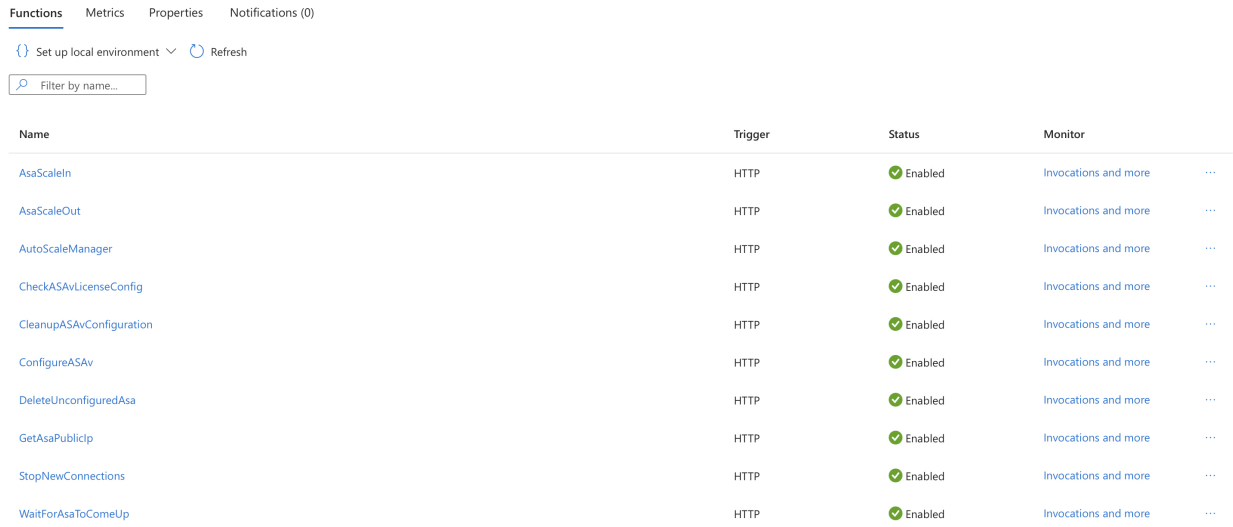
部署 ARM 模板时，Azure 会创建名称为 *<resourceNamePrefix>-function-app* 的功能应用。创建函数应用后，执行下面给出的步骤，将 Azure 集群 AutoScale 函数部署到函数应用。

## 过程

**步骤 1** 从本地计算机运行以下命令，将集群自动扩展 Azure 功能部署到函数应用。

```
az functionapp deployment source config-zip -g <Resource Group Name> -n <Function App Name>
 --src  <cluster_functions.zip> --build-remote true
```

**步骤 2** 通过检查上传的函数是否在函数应用的“**概述**”部分中可见，如下所示，验证函数部署是否成功。




# 更新 Azure 逻辑应用

逻辑应用充当Autoscale功能的协调器。ARM模板会创建一个主干逻辑应用，然后您需要手动更新，提供使之作为 Auto Scale 协调器发挥作用所需的信息。

## 过程

**步骤 1** 从存储库中将文件 *LogicApp.txt* 恢复到本地系统，然后如下所示进行编辑。

**重要事项**
在继续之前，阅读并理解所有这些步骤。

这些手动步骤不会在 ARM 模板中自动执行，以便稍后只能独立升级逻辑应用。

a) 必需： 查找所有“SUBSCRIPTION_ID”并替换为您的订用 ID 信息。
b) 必需： 查找所有“RG_NAME”并替换为您的资源组名称。
c) 必需： 查找所有“FUNCTIONAPPNAME”并替换为您的函数应用名称。

以下示例显示了 *LogicApp.txt* 文件中的几行：

```
  "AutoScaleManager": {
      "inputs": {
          "function": {
              "id":
"/subscriptions/SUBSCRIPTION_ID/resourceGroups/RG_NAME/providers/Microsoft.Web/sites/FUNCTIONAPPNAME/functions/AutoScaleManager"

          }
.
.
                      },
```

```
                    "Deploy_Changes_to_ASA": {
                        "inputs": {
                            "body": "@body('AutoScaleManager')",
                            "function": {
                                "id":
"/subscriptions/SUBSCRIPTION_ID/resourceGroups/RG_NAME/providers/Microsoft.Web/sites/FUNCTIONAPPNAME/functions/DeployConfiguration"

                            }
.
.
                    "DeviceDeRegister": {
                        "inputs": {
                            "body": "@body('AutoScaleManager')",
                            "function": {
                                "id":
"/subscriptions/SUBSCRIPTION_ID/resourceGroups/RG_NAME/providers/Microsoft.Web/sites/FUNCTIONAPPNAME/functions/DeviceDeRegister"

                            }
                        },
                        "runAfter": {
                            "Delay_For_connection_Draining": [
```

d) （可选） 编辑触发间隔，或保留默认值 (5)。这是定期触发 Autoscale 的时间间隔。以下示例显示了 *LogicApp.txt* 文件中的几行：

```
"triggers": {
    "Recurrence": {
        "conditions": [],
        "inputs": {},
        "recurrence": {
            "frequency": "Minute",
            "interval": 5
        },
```

e) （可选） 编辑要进行排空的时间，或保留默认值 (5)。这是内向扩展操作期间，在删除设备之前从 ASA 虚拟 中排空现有连接的时间间隔。以下示例显示了 *LogicApp.txt* 文件中的几行：

```
"actions": {
     "Branch_based_on_Scale-In_or_Scale-Out_condition": {
         "actions": {
             "Delay_For_connection_Draining": {
                 "inputs": {
                     "interval": {
                         "count": 5,
                         "unit": "Minute"
                     }
```

f) （可选） 编辑冷却时间，或保留默认值 (10)。这是在外向扩展完成后不执行任何操作的时间。以下示例显示了 *LogicApp.txt* 文件中的几行：

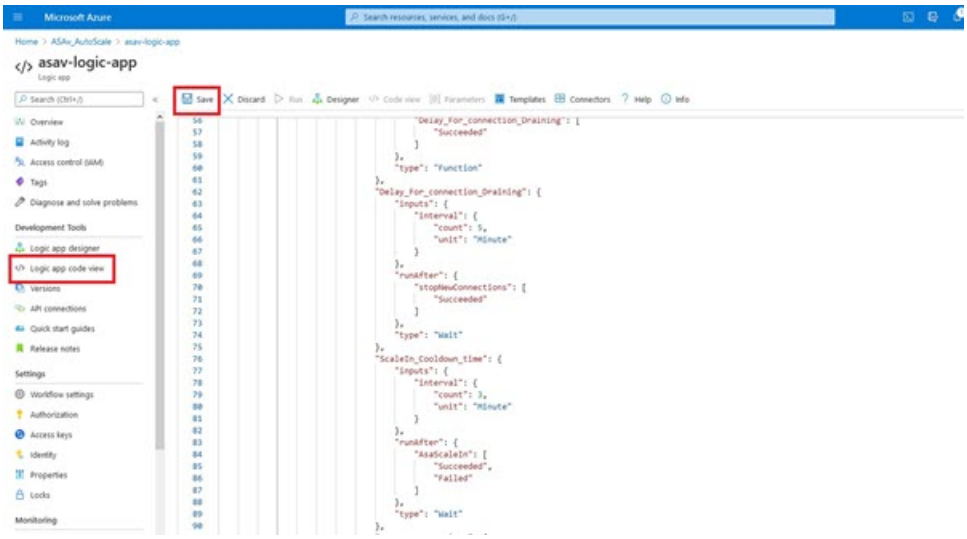
```
            "actions": {
                "Branch_based_on_Scale-Out_or_Invalid_condition": {
                    "actions": {
                        "Cooldown_time": {
                            "inputs": {
                                "interval": {
                                    "count": 10,
                                    "unit": "Second"
```

```
}
```

注释
这些步骤也可以从 Azure 门户完成。有关详细信息，请参阅 Azure 文档。

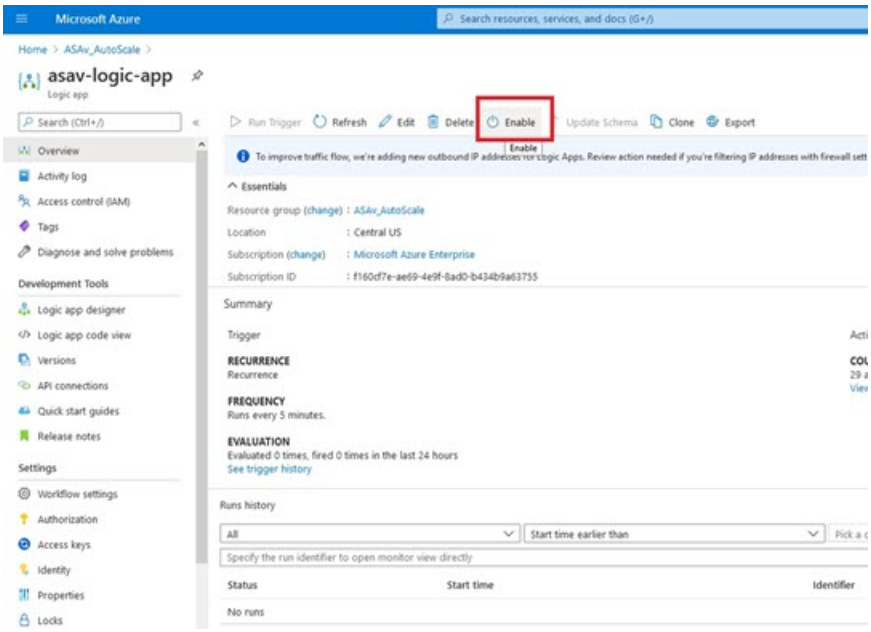
**步骤 2** 转至**逻辑应用代码视图 (Logic App code view)**，删除默认内容并粘贴编辑后的 *LogicApp.txt* 文件内容，然后点击**保存 (Save)**。

图 *8:* 逻辑应用代码视图



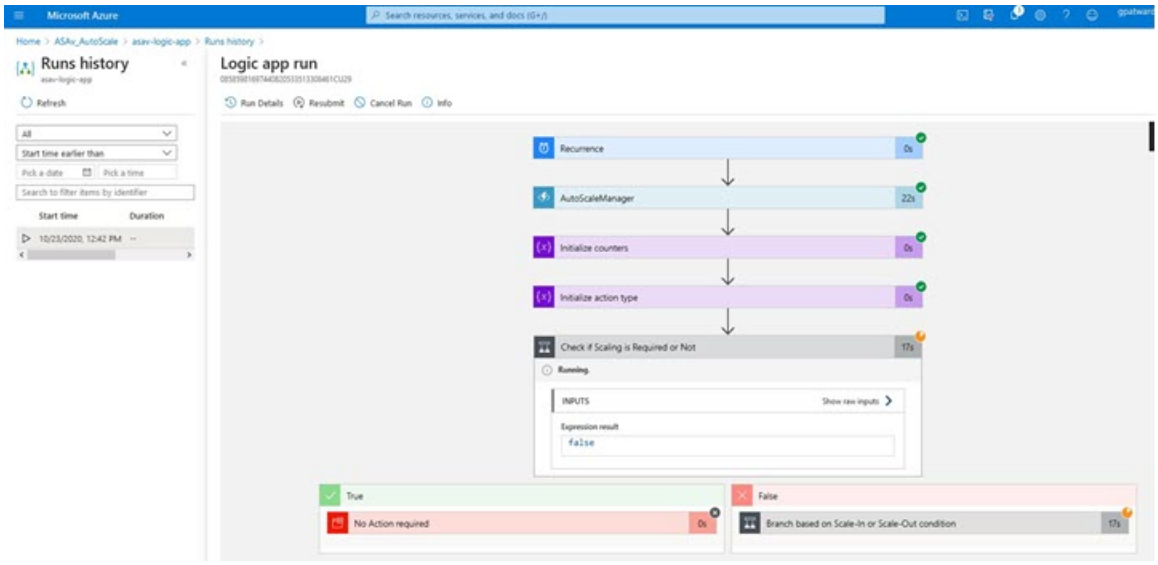

**步骤 3** 保存逻辑应用时，它处于“禁用”状态。当要启动 Auto Scale Manager 时，请点击**启用 (Enable)**。

图 *9:* 启用逻辑应用



**步骤 4** 启用后，任务就会开始运行。点击“正在运行”(Running) 状态可查看活动。

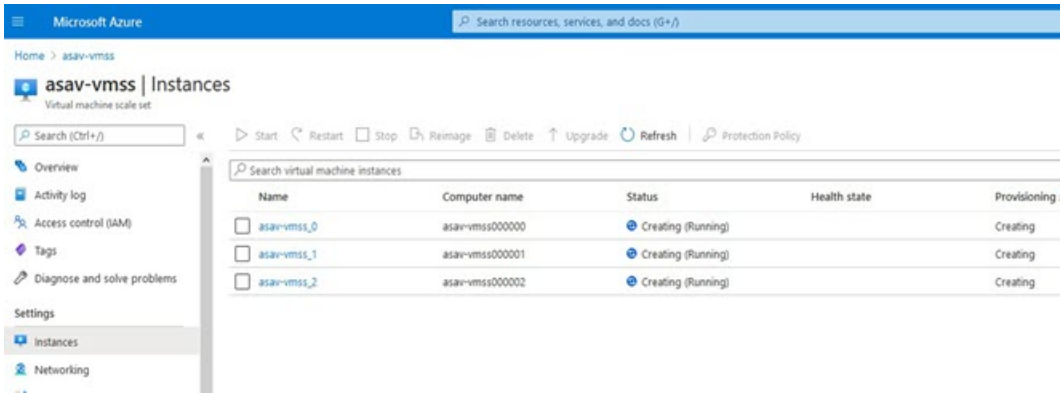
图 *10:* 逻辑应用运行状态




**步骤 5** 逻辑应用启动后，所有与部署相关的步骤都将完成。

**步骤 6** 在 VMSS 中验证是否正在创建 ASA 虚拟 实例。

图 *11: ASA Virtual* 实例运行




在此示例中，由于在 ARM 模板部署中将 'minAsaCount' 设置为“3”并将“initDeploymentMode”设置为“批量”，因此启动了三个 ASA 虚拟 实例。

# 自定义集群操作

作为第 0 天配置的一部分，或者在部署集群之后，您可以自定义集群运行状况监控、TCP 连接复制延迟、流移动性和其他优化。

在控制节点上执行这些程序。

# 配置基本 ASA 集群参数

您可以在控制节点上自定义集群设置。

过程

**步骤 1** 进入集群配置模式：

**cluster group** *name*

**步骤 2** （可选） 启用数据节点到控制节点的控制台复制：

**console-replicate**

默认情况下会禁用此功能。对于特定的重要事件，ASA 可将某些消息直接打印输出到控制台。如果启用了控制台复制，数据节点会将控制台消息发送到控制节点，因此您只需要监控集群的一个控制台端口。

**步骤 3** 设置集群事件的最低跟踪级别：

**trace-level** 级别

根据需要设置最低级别：

- **critical** - 重要事件（严重性=1）
- **warning** - 警告（严重性 = 2）
- **informational** - 信息事件（严重性=3）
- **debug** - 调试事件（严重性=4）

**步骤 4** 设置从流所有者到导向器和备份所有者的流状态刷新消息（clu_keepalive 和 clu_update 消息）的保持连接间隔。

**clu-keepalive-interval** 秒

- 秒 - 15 到 55。默认值为 15。

您可能想将间隔设置为比默认值更长的时间，以减少集群控制链路上的流量。

# 配置运行状态监控并自动重新加入设置

此程序可以配置节点和接口运行状态监控。

您可能想禁用不重要的接口（例如管理接口）的运行状况检查。运行状况监控不在 VLAN 子接口上执行。您不能为集群控制链路配置监控；它始终处于被监控状态。

过程

**步骤 1** 进入集群配置模式。

**cluster group** *name*

示例：

```
ciscoasa(config)# cluster group test
ciscoasa(cfg-cluster)#
```

**步骤 2** 自定义集群节点运行状况检查功能。

**health-check** [**holdtime** 超时]

为了确定节点运行状况，ASA 集群节点会在集群控制链路上将 heartbeat 消息发送到其他节点。如果节点在保持期内未接收到来自对等节点的任何 heartbeat 消息，则对等节点被视为无响应或无法工作。

- **holdtime** 超时 - 用于确定两次设备 heartbeat状态消息之间的时间间隔，其值介于 0.3到 45 秒；默认值为 3 秒。

当拓扑发生任何更改时（例如添加或删除数据接口、启用或禁用 ASA、或交换机上的接口），您应禁用运行状态检查功能，还要禁用对已禁用接口的接口监控 (**no health-check monitor-interface**)。当拓扑结构更改完成且配置更改已同步到所有节点后，您可以重新启用运行状况检查功能。

示例：

```
ciscoasa(cfg-cluster)# health-check holdtime 5
```

**步骤 3** 在接口上禁用接口运行状况检查。

**no health-check monitor-interface** *interface_id*

接口运行状态检查将监控链路故障。ASA 在多长时间后从集群中删除成员取决于该节点是既定成员还是正在加入集群的设备。默认情况下，为所有接口启用运行状况检查。您可以使用此命令的 **no** 形式逐个接口将其禁用。您可能想禁用不重要的接口（例如管理接口）的运行状况检查。

- *interface_id* - 禁用接口监控。运行状况监控不在 VLAN 子接口上执行。您不能为集群控制链路配置监控；它始终处于被监控状态。

当拓扑发生任何更改时（例如添加或删除数据接口、启用或禁用 ASA、或交换机上的接口），您应禁用运行状态检查功能 (**no health-check**)，还要禁用对已禁用接口的接口监控。当拓扑结构更改完成且配置更改已同步到所有节点后，您可以重新启用运行状况检查功能。

示例：

```
ciscoasa(cfg-cluster)# no health-check monitor-interface management1/1
```

**步骤 4** 自定义在运行状况检查发生故障后的自动重新加入集群设置。

**health-check** {**data-interface** | **cluster-interface** | **system**} **auto-rejoin** [**unlimited** | *auto_rejoin_max*] *auto_rejoin_interval auto_rejoin_interval_variation*

- **system**- 指定内部错误的自动重新加入设置。内部故障包括：应用程序同步超时、不一致的应用程序状态等。
- **unlimited** —（**cluster-interface** 的默认值）不限制重新加入尝试的次数。
- *auto-rejoin-max* — 设置重新加入尝试次数，介于 0 和 65535 之间。**0** 禁用自动重新加入。**data-interface** 和 **system** 的默认值为 3。
- *auto_rejoin_interval* - 定义两次重新加入尝试之间的间隔持续时间（以分钟为单位），介于 2 和 60 之间。默认值为 5 分钟。节点尝试重新加入集群的最大总时间限制为自上次失败之时起 14400 分钟（10 天）。
- *Auto_rejoin_interval_variation* - 定义是否增加间隔持续时间。设置介于 1 和 3 之间的值：**1**（无更改）；**2**（2 倍于上一次持续时间）或 **3**（3 倍于上一次持续时间）。例如，如果您将间隔持续时间设置为 5 分钟，并将变化设置为 2，则在 5 分钟后进行第 1 次尝试；在 10 分钟 (2 x 5) 后进行第 2 次尝试；在 20 分钟 (2 x 10) 后进行第 3 次尝试。对于集群接口，默认值为 **1**，而对于数据接口和系统，默认值为 **2**。

示例：

```
ciscoasa(cfg-cluster)# health-check data-interface auto-rejoin 10 3 3
```

**步骤 5** 配置 ASA 将接口视为发生故障并将节点从集群中删除之前经过的防反跳时间。

**health-check monitor-interface debounce-time** *ms*

示例：

```
ciscoasa(cfg-cluster)# health-check monitor-interface debounce-time 300
```

将防反跳时间设置为 300 到 9000 毫秒之间。默认值为 500 毫秒。较小的值可以加快检测接口故障的速度。请注意，如果配置的防反跳时间较低，会增加误报几率。在发生接口状态更新时，ASA 会等待指定的毫秒数，然后才将接口标记为发生故障，并将节点从集群中删除。

**步骤 6** （可选） 配置流量负载监控。

**load-monitor** [ **frequency** *seconds*] [ **intervals** *intervals*]

- **frequency** *seconds* — 设置监控消息之间的时间（以秒为单位），范围介于 10 到 360 秒之间。默认值为 20 秒。
- **intervals** *intervals* — 设置 ASA 维护数据的间隔的数量，该值介于 1 到 60 之间。默认值为 30。

您可以监控集群成员的流量负载，包括总连接计数、CPU 和内存使用情况以及缓冲区丢弃。如果负载过高，且剩余的节点可以处理负载，您可以选择在节点上手动禁用集群，或调整外部交换机上的负载均衡。默认情况下启用此功能。您可以定期监控流量负载。如果负载过高，您可以选择手动禁用节点上的集群。

使用 **show cluster info load-monitor** 命令查看流量负载。

示例：

```
ciscoasa(cfg-cluster)# load-monitor frequency 50 intervals 25
ciscoasa(cfg-cluster)# show cluster info load-monitor
ID  Unit Name
0  B
1  A_1
Information from all units with 50 second interval:
Unit     Connections    Buffer Drops   Memory Used   CPU Used
Average from last 1 interval:
   0           0            0            14              25
   1           0            0            16              20
Average from last 25 interval:
   0           0            0            12              28
   1           0            0            13              27
```

示例

以下示例将 health-check holdtime 配置为 0.3 秒；禁用GUANLI 0/0 接口上的监控；将数据接口的 auto-rejoin 设置为从 2 分钟开始的 4 次尝试，将 duration 增至上一次间隔的 3 倍；以及将集群控制链路的 auto-rejoin 设为 6 次尝试，每隔 2 分钟一次。

```
ciscoasa(config)# cluster group test
ciscoasa(cfg-cluster)# health-check holdtime .3
ciscoasa(cfg-cluster)# no health-check monitor-interface management0/0
ciscoasa(cfg-cluster)# health-check data-interface auto-rejoin 4 2 3
ciscoasa(cfg-cluster)# health-check cluster-interface auto-rejoin 6 2 1
```

# 管理集群节点

部署集群后，您可以更改配置和管理集群节点。

## 成为非活动节点

要成为集群的非活动成员，请在节点上禁用集群，同时保持集群配置不变。

注释 当ASA处于非活动状态（以手动方式或因运行状况检查失败）时，所有数据接口都将关闭；只有管理专用接口可以发送和接收流量。要恢复流量传输，请重新启用集群；或者，您也可以从集群中完全删除该节点。管理接口将保持打开，使用节点从集群 IP 池接收的 IP 地址。但如果您重新加载，而节点仍在集群中处于非主用状态（例如，您保存了已禁用集群的配置），则管理接口将被禁用。您必须使用控制台端口来进行任何进一步配置。

过程

**步骤 1** 进入集群配置模式：

**cluster group** *name*

示例：

```
ciscoasa(config)# cluster group pod1
```

**步骤 2** 禁用集群：

**no enable**

如果此节点是控制节点，则会进行新的控制选择，并且其他成员将成为控制节点。

集群配置保持不变，因此您可于稍后再次启用集群。

# 从控制节点停用数据节点

要禁用您登录的节点以外的成员，请执行以下步骤。

注释 当ASA处于非活动状态时，所有数据接口关闭；只有管理专用接口可以发送和接收流量。要恢复流量流，请重新启用集群。管理接口将保持打开，使用节点从集群 IP 池接收的 IP 地址。但如果您重新加载，而节点仍在集群中处于非主用状态（例如，假设您保存了已禁用集群的配置），管理接口将被禁用。您必须使用控制台端口来进行任何进一步的配置。

过程

从集群中删除该节点：

**cluster remove unit** *node_name*

引导程序配置保持不变，从控制节点同步的最新配置也保持不变，因此您可于稍后重新添加该节点而不会丢失配置。如果在数据节点上输入此命令来删除控制节点，则会选择新的控制节点。

要查看成员名称，请输入 **cluster remove unit ?**，或者输入 **show cluster info** 命令。

示例：

```
ciscoasa(config)# cluster remove unit ?

Current active units in the cluster:
asa2
```

```
ciscoasa(config)# cluster remove unit asa2
WARNING: Clustering will be disabled on unit asa2. To bring it back
to the cluster please logon to that unit and re-enable clustering
```

# 重新加入集群

如果从集群中删除了某个节点（例如针对出现故障的接口），或者如果您手动停用了某个成员，那么您必须手动重新加入集群。

过程

**步骤 1** 在控制台中，进入集群配置模式：

**cluster group** *name*

示例：

```
ciscoasa(config)# cluster group pod1
```

**步骤 2** 启用集群。

**enable**

# 离开集群

要完全退出集群，需要删除整个集群引导程序配置。由于每个节点上的当前配置相同（从主用设备同步），因此退出集群就意味着要么从备份恢复集群前的配置，要么清除现有配置并重新开始配置以免 IP 地址冲突。

过程

**步骤 1** 对于数据节点，禁用集群：

**cluster group** *cluster_name* **no enable**

示例：

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)# no enable
```

在数据节点上启用集群时，您无法进行配置更改。

**步骤 2** 清除集群配置：

**clear configure cluster**

ASA 将关闭所有接口，包括管理接口和集群控制链路。

**步骤 3** 禁用集群接口模式：

**no cluster interface-mode**

模式并非存储于配置中，因此必须手动重置。

**步骤 4** 如果有备份配置，可将备份配置复制到正在运行的配置中：

**copy** *backup_cfg* **running-config**

示例：

```
ciscoasa(config)# copy backup_cluster.cfg running-config

Source filename [backup_cluster.cfg]?

Destination filename [running-config]?
ciscoasa(config)#
```

**步骤 5** 将配置保存到启动配置：

**write memory**

**步骤 6** 如果没有备份配置，请重新配置管理访问。例如，确保更改接口 IP 地址，并恢复正确的主机名。

# 更改控制节点

⚠

**注意** 要更改控制节点，最好的方法是在控制节点上禁用集群，等到新的控制选举后再重新启用集群。如果必须指定要成为控制节点的具体节点，请使用本节中的程序。但是请注意，对集中功能而言，如果使用本程序强制更改控制节点，则所有连接都将断开，而您必须在新的控制节点上重新建立连接。

要更改控制节点，请执行以下步骤：

过程

将新节点设置为控制节点：

**cluster control-node unit***node_name*

示例：

```
ciscoasa(config)# cluster control-node unit asa2
```

您需要重新连接到主集群 IP 地址。

要查看成员名称，请输入 **cluster control-node unit ?**（可查阅除当前节点之外的所有名称），或输入 **show cluster info** 命令。

# 在整个集群范围内执行命令

要向集群中的所有节点或某个特定节点发送命令，请执行以下步骤。向所有节点发送 **show** 命令以收集所有输出并将其显示在当前节点的控制台上。也可在整个集群范围内执行其他命令（如 **capture** 和 **copy**）。

## 过程

发送命令到所有节点，或者如果指定了节点名称，则发送到特定节点：

**cluster exec** [**unit** *node_name*] *command*

**示例：**

```
ciscoasa# cluster exec show xlate
```

要查看节点名称，请输入 **cluster exec unit ?**（可查阅除当前节点之外的所有名称），或输入 **show cluster info** 命令。

### 示例

要同时将同一捕获文件从集群中的所有节点复制到 TFTP 服务器，请在控制节点上输入以下命令：

```
ciscoasa# cluster exec copy /pcap capture: tftp://10.1.1.56/capture1.pcap
```

多个 PCAP 文件（一个文件来自一个节点）将复制到 TFTP 服务器。目标捕获文件名会自动附加节点名称，例如 capture1_asa1.pcap、capture1_asa2.pcap 等。在本例中，asa1和asa2是集群节点名称。

# 监控集群

您可以监控集群状态和连接并排除故障。

# 监控集群状态

请参阅以下命令来监控集群状态：

- **show cluster info** [**health** [**details**]]

  如果没有关键字，**show cluster info** 命令将显示所有集群成员的状态。

  **show cluster info health** 命令将显示接口、节点和整个集群的当前运行状况。**details** 关键字显示心跳消息失败的次数。

  请参阅 **show cluster info** 命令的以下输出：

```
ciscoasa# show cluster info
Cluster stbu: On
  This is "C" in state DATA_NODE
      ID        : 0
      Site ID   : 1
            Version   : 9.4(1)
      Serial No.: P3000000025
      CCL IP    : 10.0.0.3
      CCL MAC   : 000b.fcf8.c192
      Last join : 17:08:59 UTC Sep 26 2011
      Last leave: N/A
Other members in the cluster:
  Unit "D" in state DATA_NODE
      ID        : 1
      Site ID   : 1
            Version   : 9.4(1)
      Serial No.: P3000000001
      CCL IP    : 10.0.0.4
      CCL MAC   : 000b.fcf8.c162
      Last join : 19:13:11 UTC Sep 23 2011
      Last leave: N/A
  Unit "A" in state CONTROL_NODE
      ID        : 2
      Site ID   : 2
            Version   : 9.4(1)
      Serial No.: JAB0815R0JY
      CCL IP    : 10.0.0.1
      CCL MAC   : 000f.f775.541e
      Last join : 19:13:20 UTC Sep 23 2011
      Last leave: N/A
  Unit "B" in state DATA_NODE
      ID        : 3
      Site ID   : 2
            Version   : 9.4(1)
      Serial No.: P3000000191
      CCL IP    : 10.0.0.2
      CCL MAC   : 000b.fcf8.c61e
      Last join : 19:13:50 UTC Sep 23 2011
      Last leave: 19:13:36 UTC Sep 23 2011
```

- **show cluster info auto-join**

  显示集群节点是否将在一段延迟后自动重新加入集群，以及是否已清除故障条件（例如等待许可证、机箱运行状况检查失败，等等）。如果节点已永久禁用，或节点已在集群中，则此命令将不会显示任何输出。

  请参阅 **show cluster info auto-join** 命令的以下输出：

```
ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster in 253 seconds.
Quit reason: Received control message DISABLE

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Control node has application down that data node has up.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Chassis-blade health check failed.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Service chain application became down.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Unit is kicked out from cluster because of Application health check failure.


ciscoasa(cfg-cluster)# show cluster info auto-join
Unit join is pending (waiting for the smart license entitlement: ent1)

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit join is pending (waiting for the smart license export control flag)
```

- **show cluster info transport**{**asp** |**cp** [**detail**]}

  显示以下项目传输相关的统计信息：

  - **asp** — 数据平面传输统计信息。
  - **cp** — 控制平面传输统计信息。

    如果您输入 **detail** 关键字，您可以查看集群可靠传输协议的使用情况，以便确定控制平面中的缓冲区已满时的丢包问题。请参阅 **show cluster info transport cp detail** 命令的以下输出：

```
ciscoasa# show cluster info transport cp detail
Member ID to name mapping:
  0 - unit-1-1   2 - unit-4-1   3 - unit-2-1

Legend:
  U    - unreliable messages
  UE   - unreliable messages error
  SN   - sequence number
  ESN  - expecting sequence number
  R    - reliable messages
  RE   - reliable messages error
  RDC  - reliable message deliveries confirmed
  RA   - reliable ack packets received
  RFR  - reliable fast retransmits
  RTR  - reliable timer-based retransmits
  RDP  - reliable message dropped
  RDPR - reliable message drops reported
  RI   - reliable message with old sequence number
  RO   - reliable message with out of order sequence number
  ROW  - reliable message with out of window sequence number
  ROB  - out of order reliable messages buffered
```

```
  RAS  - reliable ack packets sent

This unit as a sender
-------------------------
       all       0         2         3
  U    123301    3867966   3230662   3850381
  UE   0         0         0         0
  SN   1656a4ce  acb26fe   5f839f76  7b680831
  R    733840    1042168   852285    867311
  RE   0         0         0         0
  RDC  699789    934969    740874    756490
  RA   385525    281198    204021    205384
  RFR  27626     56397     0         0
  RTR  34051     107199    111411    110821
  RDP  0         0         0         0
  RDPR 0         0         0         0

This unit as a receiver of broadcast messages
----------------------------------------------
       0         2         3
  U    111847    121862    120029
  R    7503      665700    749288
  ESN  5d75b4b3  6d81d23   365ddd50
  RI   630       34278     40291
  RO   0         582       850
  ROW  0         566       850
  ROB  0         16        0
  RAS  1571      123289    142256

This unit as a receiver of unicast messages
-----------------------------------------------
       0         2         3
  U    1         3308122   4370233
  R    513846    879979    1009492
  ESN  4458903a  6d841a84  7b4e7fa7
  RI   66024     108924    102114
  RO   0         0         0
  ROW  0         0         0
  ROB  0         0         0
  RAS  130258    218924    228303

Gated Tx Buffered Message Statistics
------------------------------------
    current sequence number: 0

    total:                   0
    current:                 0
    high watermark:          0

    delivered:               0
    deliver failures:        0

    buffer full drops:       0
    message truncate drops:  0

    gate close ref count:    0

    num of supported clients:45

MRT Tx of broadcast messages
============================
Message high watermark: 3%
  Total messages buffered at high watermark: 5677
  [Per-client message usage at high watermark]
```

```
  ------------------------------------------------------------
  Client name                          Total messages  Percentage
  Cluster Redirect Client                        4153         73%
  Route Cluster Client                            419          7%
  RRI Cluster Client                             1105         19%

Current MRT buffer usage: 0%
  Total messages buffered in real-time: 1
  [Per-client message usage in real-time]
  Legend:
        F - MRT messages sending when buffer is full
        L - MRT messages sending when cluster node leave
        R - MRT messages sending in Rx thread
  ------------------------------------------------------------------------------
  Client name                          Total messages  Percentage    F   L   R
  VPN Clustering HA Client                          1        100%    0   0   0

MRT Tx of unitcast messages(to member_id:0)
==========================================
Message high watermark: 31%
  Total messages buffered at high watermark: 4059
  [Per-client message usage at high watermark]
  ------------------------------------------------------------
  Client name                          Total messages  Percentage
  Cluster Redirect Client                        3731         91%
  RRI Cluster Client                              328          8%

Current MRT buffer usage: 29%
  Total messages buffered in real-time: 3924
  [Per-client message usage in real-time]
  Legend:
        F - MRT messages sending when buffer is full
        L - MRT messages sending when cluster node leave
        R - MRT messages sending in Rx thread
  ------------------------------------------------------------------------------
  Client name                          Total messages  Percentage    F   L   R
  Cluster Redirect Client                        3607         91%    0   0   0
  RRI Cluster Client                              317          8%    0   0   0

MRT Tx of unitcast messages(to member_id:2)
==========================================
Message high watermark: 14%
  Total messages buffered at high watermark: 578
  [Per-client message usage at high watermark]
  ------------------------------------------------------------
  Client name                          Total messages  Percentage
  VPN Clustering HA Client                        578        100%

Current MRT buffer usage: 0%
  Total messages buffered in real-time: 0

MRT Tx of unitcast messages(to member_id:3)
==========================================
Message high watermark: 12%
  Total messages buffered at high watermark: 573
  [Per-client message usage at high watermark]
  ------------------------------------------------------------
  Client name                          Total messages  Percentage
  VPN Clustering HA Client                        572         99%
  Cluster VPN Unique ID Client                      1          0%

Current MRT buffer usage: 0%
  Total messages buffered in real-time: 0
```

- **show cluster history**

  显示集群历史记录，以及有关集群节点加入失败的原因或节点离开集群的原因的错误消息。

# 捕获整个集群范围内的数据包

有关在集群中捕获数据包的信息，请参阅以下命令：

**cluster exec capture**

要支持集群范围的故障排除，您可以使用 **cluster exec capture** 命令在控制节点上启用捕获集群特定流量的功能，随后集群中的所有数据节点上将自动启用此功能。

# 监控集群资源

请参阅以下命令以监控集群资源：

**show cluster** {**cpu** | **memory** | **resource**} [*options*]

显示整个集群的聚合数据。可用 *options* 取决于数据类型。

# 监控集群流量

请参阅以下命令以监控集群流量：

- **show conn [detail], cluster exec show conn**

  **show conn** 命令显示一个传输是导向者、备用还是转发者传输。在任意节点上使用 **cluster exec show conn** 命令可查看所有连接。此命令可以显示单个流的流量到达集群中不同 ASA 的方式。集群的吞吐量取决于负载均衡的效率和配置。此命令可以让您很方便地查看某个连接的流量如何流经集群，也可以帮助您了解负载均衡器对传输的性能有何影响。

  **show conn detail** 命令还显示哪些流应 遵守流移动性。

  以下是 **show conn detail** 命令的输出示例：

```
ciscoasa/ASA2/data node# show conn detail
12 in use, 13 most used
Cluster stub connections: 0 in use, 46 most used
Flags: A - awaiting inside ACK to SYN, a - awaiting outside ACK to SYN,
       B - initial SYN from outside, b - TCP state-bypass or nailed,
       C - CTIQBE media, c - cluster centralized,
       D - DNS, d - dump, E - outside back connection, e - semi-distributed,
       F - outside FIN, f - inside FIN,
       G - group, g - MGCP, H - H.323, h - H.225.0, I - inbound data,
       i - incomplete, J - GTP, j - GTP data, K - GTP t3-response
       k - Skinny media, L - LISP triggered flow owner mobility,
       M - SMTP data, m - SIP media, n - GUP
       O - outbound data, o - offloaded,
       P - inside back connection,
       Q - Diameter, q - SQL*Net data,
       R - outside acknowledged FIN,
       R - UDP SUNRPC, r - inside acknowledged FIN, S - awaiting inside SYN,
       s - awaiting outside SYN, T - SIP, t - SIP transient, U - up,
```

```
       V - VPN orphan, W - WAAS,
       w - secondary domain backup,
       X - inspected by service module,
       x - per session, Y - director stub flow, y - backup stub flow,
       Z - Scansafe redirection, z - forwarding stub flow
ESP outside: 10.1.227.1/53744 NP Identity Ifc: 10.1.226.1/30604, , flags c, idle 0s,
uptime
1m21s, timeout 30s, bytes 7544, cluster sent/rcvd bytes 0/0, owners (0,255) Traffic
received
at interface outside Locally received: 7544 (93 byte/s) Traffic received at interface
NP
Identity Ifc Locally received: 0 (0 byte/s) UDP outside: 10.1.227.1/500 NP Identity
Ifc:
10.1.226.1/500, flags -c, idle 1m22s, uptime 1m22s, timeout 2m0s, bytes 1580, cluster
sent/rcvd bytes 0/0, cluster sent/rcvd total bytes 0/0, owners (0,255) Traffic received
 at
interface outside Locally received: 864 (10 byte/s) Traffic received at interface NP
Identity
Ifc Locally received: 716 (8 byte/s)
```

要对连接流进行故障排除，请先在任意节点上输入 **cluster exec show conn** 命令查看所有节点上的连接。寻找具有以下标志的流：导向者 (Y)、备用 (y) 和转发者 (z)。下例显示了三台 ASA 上的一条从 172.18.124.187:22 到 192.168.103.131:44727 的 SSH 连接；ASA 1 带有 z 标志，表示其是该连接的转发者；ASA3 带有 Y 标志，表示其是该连接的导向者；而 ASA2 则没有特殊的标志，表示其是所有者。在出站方向，此连接的数据包进入 ASA2 上的内部接口并从外部接口流出。在入站方向，此连接的数据包进入 ASA 1 和 ASA3 上的外部接口，通过集群控制链路被转发到 ASA2，然后流出 ASA2 上的内部接口。

```
ciscoasa/ASA1/control node# cluster exec show conn
ASA1(LOCAL):**************************************************************
18 in use, 22 most used
Cluster stub connections: 0 in use, 5 most used
TCP outside  172.18.124.187:22 inside  192.168.103.131:44727, idle 0:00:00, bytes
37240828, flags z


ASA2:**************************************************************************
12 in use, 13 most used
Cluster stub connections: 0 in use, 46 most used
TCP outside  172.18.124.187:22 inside  192.168.103.131:44727, idle 0:00:00, bytes
37240828, flags UIO


ASA3:**************************************************************************
10 in use, 12 most used
Cluster stub connections: 2 in use, 29 most used
TCP outside  172.18.124.187:22 inside  192.168.103.131:44727, idle 0:00:03, bytes 0,
flags  Y
```

- **show cluster info** [**conn-distribution** | **packet-distribution** | **loadbalance** | **flow-mobility counters**]

  **show cluster info conn-distribution** 和 **show cluster info packet-distribution** 命令显示流量在所有集群节点上的分布。这些命令可以帮助您评估和调整外部负载均衡器。

  **show cluster info loadbalance** 命令显示连接再均衡统计信息。

  The **show cluster info flow-mobility counters** 命令显示 EID 移动和流所有者移动信息。请参阅 **show cluster info flow-mobility counters** 的以下输出：

```
ciscoasa# show cluster info flow-mobility counters
EID movement notification received  : 4
EID movement notification processed : 4
Flow owner moving requested         : 2
```

- **show cluster info load-monitor** [**details**]

  **show cluster info load-monitor** 命令显示最后一个间隔的集群成员的流量负载，以及已配置的总间隔数（默认情况下为 30）。使用 **details** 关键字查看每个时间间隔的每个度量值。

```
ciscoasa(cfg-cluster)# show cluster info load-monitor
ID  Unit Name
0  B
1  A_1
Information from all units with 20 second interval:
Unit     Connections    Buffer Drops   Memory Used   CPU Used
Average from last 1 interval:
   0           0            0            14              25
   1           0            0            16              20
Average from last 30 interval:
   0           0            0            12              28
   1           0            0            13              27
```

```
ciscoasa(cfg-cluster)# show cluster info load-monitor details


ID  Unit Name

 0  B

 1  A_1

Information from all units with 20 second interval


Connection count captured over 30 intervals:

Unit ID  0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

Unit ID  1

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0
```

```
         0          0          0          0          0          0

Buffer drops captured over 30 intervals:
Unit ID  0
         0          0          0          0          0          0
         0          0          0          0          0          0
         0          0          0          0          0          0
         0          0          0          0          0          0
         0          0          0          0          0          0
Unit ID  1
         0          0          0          0          0          0
         0          0          0          0          0          0
         0          0          0          0          0          0
         0          0          0          0          0          0
         0          0          0          0          0          0

Memory usage(%) captured over 30 intervals:
Unit ID  0
        25         25         30         30         30         35
        25         25         35         30         30         30
        25         25         30         25         25         35
        30         30         30         25         25         25
        25         20         30         30         30         30
Unit ID  1
        30         25         35         25         30         30
        25         25         35         25         30         35
        30         30         35         30         30         30
        25         20         30         25         25         30
        20         30         35         30         30         35

CPU usage(%) captured over 30 intervals:
Unit ID  0
```

```
        25          25          30          30          30          35
        25          25          35          30          30          30
        25          25          30          25          25          35
        30          30          30          25          25          25
        25          20          30          30          30          30
Unit ID  1
        30          25          35          25          30          30
        25          25          35          25          30          35
        30          30          35          30          30          30
        25          20          30          25          25          30
        20          30          35          30          30          35
```

- **show cluster** {**access-list** | **conn** | **traffic** | **user-identity** | **xlate**} [*options*]

  显示整个集群的聚合数据。可用 *options* 取决于数据类型。

  请参阅 **show cluster access-list** 命令的以下输出：

```
ciscoasa# show cluster access-list
hitcnt display order: cluster-wide aggregated result, unit-A, unit-B,  unit-C, unit-D
access-list cached ACL log flows: total 0, denied 0 (deny-flow-max 4096) alert-interval
 300
access-list 101; 122 elements; name hash: 0xe7d586b5
access-list 101 line 1 extended permit tcp 192.168.143.0 255.255.255.0 any eq www
(hitcnt=0, 0, 0, 0, 0) 0x207a2b7d
access-list 101 line 2 extended permit tcp any 192.168.143.0 255.255.255.0 (hitcnt=0,
0, 0, 0, 0) 0xfe4f4947
access-list  101 line 3 extended permit tcp host 192.168.1.183 host 192.168.43.238
(hitcnt=1, 0, 0, 0, 1) 0x7b521307
access-list 101 line 4 extended permit tcp host 192.168.1.116 host 192.168.43.238
(hitcnt=0, 0, 0, 0, 0) 0x5795c069
access-list 101 line 5 extended permit tcp host 192.168.1.177 host 192.168.43.238
(hitcnt=1, 0, 0, 1, 0) 0x51bde7ee
access list 101 line 6 extended permit tcp host 192.168.1.177 host 192.168.43.13
(hitcnt=0, 0, 0, 0, 0) 0x1e68697c
access-list 101 line 7 extended permit tcp host 192.168.1.177 host 192.168.43.132
(hitcnt=2, 0, 0, 1, 1) 0xc1ce5c49
access-list 101 line 8 extended permit tcp host 192.168.1.177 host 192.168.43.192
(hitcnt=3, 0, 1, 1, 1) 0xb6f59512
access-list 101 line 9 extended permit tcp host 192.168.1.177 host 192.168.43.44
(hitcnt=0, 0, 0, 0, 0) 0xdc104200
access-list 101 line 10 extended permit tcp host 192.168.1.112 host 192.168.43.44
(hitcnt=429, 109, 107, 109, 104)
0xce4f281d
access-list 101 line 11 extended permit tcp host 192.168.1.170 host 192.168.43.238
(hitcnt=3, 1, 0, 0, 2) 0x4143a818
access-list 101 line 12 extended permit tcp host 192.168.1.170 host 192.168.43.169
(hitcnt=2, 0, 1, 0, 1) 0xb18dfea4
access-list 101 line 13 extended permit tcp host 192.168.1.170 host 192.168.43.229
(hitcnt=1, 1, 0, 0, 0) 0x21557d71
access-list 101 line 14 extended permit tcp host 192.168.1.170 host 192.168.43.106
(hitcnt=0, 0, 0, 0, 0) 0x7316e016
```

```
access-list 101 line 15 extended permit tcp host 192.168.1.170 host 192.168.43.196
(hitcnt=0, 0, 0, 0, 0) 0x013fd5b8
access-list 101 line 16 extended permit tcp host 192.168.1.170 host 192.168.43.75
(hitcnt=0, 0, 0, 0, 0) 0x2c7dba0d
```

要显示所有节点在用连接的汇聚计数，请输入：

```
ciscoasa# show cluster conn count
Usage Summary In Cluster:*********************************************
  200 in use (cluster-wide aggregated)
     cl2(LOCAL):**************************************************************
  100 in use, 100 most used

  cl1:*********************************************************************
  100 in use, 100 most used
```

- **show asp cluster counter**

  此命令对于数据路径故障排除非常有用。

# 监控集群路由

有关集群路由的信息，请参阅以下命令：

- **show route cluster**
- **debug route cluster**

  显示集群的路由信息。

- **show lisp eid**

  显示 ASA EID 表，表中显示了 EID 和站点 ID。

  请参阅 **cluster exec show lisp eid** 命令的以下输出。

```
ciscoasa# cluster exec show lisp eid
L1(LOCAL):***************************************************************
     LISP EID       Site ID
    33.44.33.105        2
    33.44.33.201        2
    11.22.11.1             4
    11.22.11.2             4
L2:**************************************************************************
     LISP EID       Site ID
    33.44.33.105        2
    33.44.33.201        2
    11.22.11.1   4
    11.22.11.2  4
```

- **show asp table classify domain inspect-lisp**

  该命令对于故障排除非常有用。

# 配置集群日志记录

有关为集群配置日志记录的信息，请参阅以下命令：

**logging device-id**

集群中的每个节点将独立生成系统日志消息。您可以使用 **logging device-id** 命令来生成具有相同或不同设备 ID 的系统日志消息，以使消息看上去是来自集群中的相同或不同节点。

# 监控集群接口

请参阅以下用于监控集群接口的命令：

- **show cluster interface-mode**

  显示集群接口模式。

# 调试集群

请参阅以下用于调试集群的命令：

- **debug cluster [ccp | datapath | fsm | general | hc | license | rpc | transport]**

  显示集群的调试消息。

- **debug cluster flow-mobility**

  显示与集群流移动性相关的事件。

- **debug lisp eid-notify-intercept**

  当 eid-notify 被拦截时显示事件。

- **show cluster info trace**

  **show cluster info trace** 命令显示调试信息，供进一步排除故障之用。

  请参阅 **show cluster info trace** 命令的以下输出：

```
ciscoasa# show cluster info trace
 Feb 02 14:19:47.456 [DBUG]Receive CCP message: CCP_MSG_LOAD_BALANCE
 Feb 02 14:19:47.456 [DBUG]Receive CCP message: CCP_MSG_LOAD_BALANCE
 Feb 02 14:19:47.456 [DBUG]Send CCP message to all: CCP_MSG_KEEPALIVE from 80-1 at
CONTROL_NODE
```

  例如，如果您看到以下消息显示两个具有相同 **local-unit** 名称的节点充当控制节点，这可能意味着两个节点具有相同的 **local-unit** 名称（请检查您的配置），或者某个节点正在接收自己的广播消息（请检查您的网络）。

```
ciscoasa# show cluster info trace
 May 23 07:27:23.113 [CRIT]Received datapath event 'multi control_nodes' with parameter
 1.
 May 23 07:27:23.113 [CRIT]Found both unit-9-1 and unit-9-1 as control_node units.
```

```
Control_node role retained by unit-9-1, unit-9-1 will leave then join as a Data_node
 May 23 07:27:23.113 [DBUG]Send event (DISABLE, RESTART | INTERNAL-EVENT, 5000 msecs,
Detected another Control_node, leave and re-join as Data_node) to FSM. Current state
CONTROL_NODE
 May 23 07:27:23.113 [INFO]State machine changed from state CONTROL_NODE to DISABLED
```

# 集群参考

本部分包括有关集群工作原理的详细信息。

## ASA 功能和集群

部分 ASA 功能不受 ASA 集群支持，还有部分功能仅在控制节点上受支持。其他功能可能对如何正确使用规定了注意事项。

### 集群不支持的功能

以下功能在启用集群的情况下无法配置，相关命令会被拒绝。

- 依靠 TLS 代理实现的统一通信功能
- 远程访问 VPN（SSL VPN 和 IPsec VPN）
- 虚拟隧道接口 (VTI)
- 以下应用检查：
  - CTIQBE
  - H323、H225 和 RAS
  - IPsec 穿透
  - MGCP
  - MMP
  - RTSP
  - SCCP（瘦客户端）
  - WAAS
  - WCCP
- 僵尸网络流量过滤器
- 自动更新服务器
- DHCP 客户端、服务器和代理。支持 DHCP 中继。
- VPN 负载均衡

- Azure 上的故障转移
- 集成路由和桥接
- FIPS 型号

## 集群集中化功能

以下功能只有在控制节点上才受支持，且无法为集群扩展。

注释 集中功能的流量从成员节点通过集群控制链路转发到控制节点。

如果使用再均衡功能，则会先将集中功能的流量再均衡到非控制节点的设备，然后再将该流量归类为集中功能；如果发生此情况，该流量随后将被发送回控制节点。

对集中功能而言，如果控制节点发生故障，则所有连接都将断开，而您必须在新的控制节点上重新建立连接。

- 以下应用检查：
  - DCERPC
  - ESMTP
  - IM
  - NetBIOS
  - PPTP
  - RADIUS
  - RSH
  - SNMP
  - SQLNET
  - SUNRPC
  - TFTP
  - XDMCP
- 静态路由监控
- 网络访问的身份验证和授权。记帐被分散。
- 筛选服务
- 站点间 VPN
- 组播路由

## 应用到单个节点的功能

这些功能将应用到每个 ASA 节点而非整个集群或控制节点。

- QoS - QoS 策略将于配置复制过程中在集群中同步。但是，该策略是在每个节点上独立执行。例如，如果对输出配置管制，则要对流出特定 ASA 的流量应用符合规则的速率和符合规则的突发量值。在包含 3 个节点且流量均衡分布的集群中，符合规则的速率实际上变成了集群速率的 3 倍。
- 威胁检测 - 威胁检测在各节点上独立工作；例如，排名统计信息就要视具体节点而定。以端口扫描检测为例，由于扫描的流量将在所有节点间进行负载均衡，而一个节点无法看到所有流量，因此端口扫描检测无法工作。

## 用于网络访问的 AAA 和集群

用于网络访问的 AAA 由三部分组成：身份验证、授权和记账。身份验证和授权作为集中功能在集群控制节点上实施，且数据结构复制到集群数据节点。如果选举出控制节点，新的控制节点将拥有所需的所有信息，以继续不间断运行经过验证的既有用户及其相关授权。当控制节点发生变化时，用户身份验证的空闲和绝对超时将保留。

记账作为分散的功能在集群中实施。记账按每次流量完成，因此在为流量配置记账时，作为流量所有者的集群节点会将记账开始和停止消息发送到 AAA 服务器。

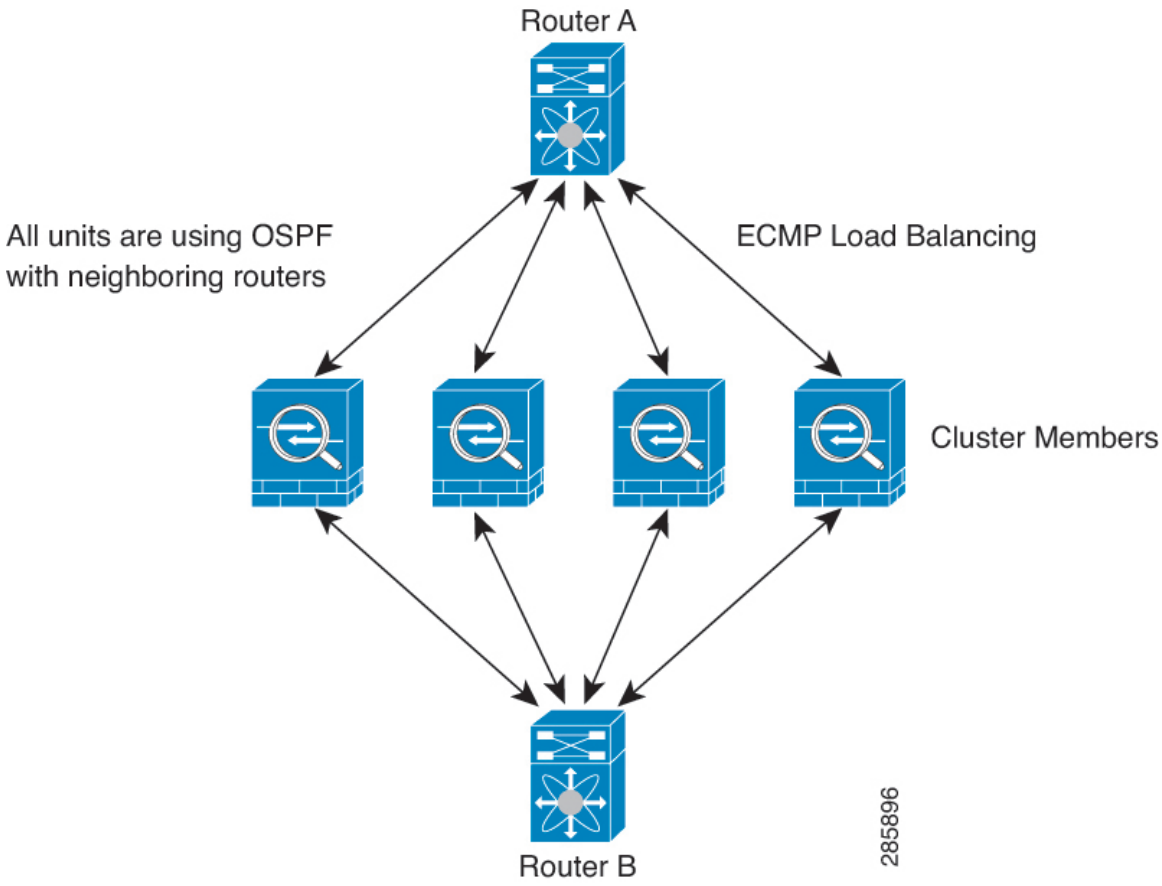
## 连接设置和集群

连接限制在集群范围强制实施（请参阅 **set connection conn-max**、**set connection embryonic-conn-max**、**set connection per-client-embryonic-max** 和 **set connection per-client-max** 命令）。每个节点都有根据广播消息估计的集群范围的计数器值。出于效率考虑，在集群中配置的连接限制可能不会严格按限制数量实施。每个节点在任何指定时间都可能高估或低估集群范围内的计数器值。不过，在负载均衡的集群中，该信息将随时间而更新。

## 动态路由和集群

在独立接口模式下，每个节点作为独立的路由器运行路由协议，且每个节点独立获知路由。

图 *12:* 独立接口模式下的动态路由




在上图中，路由器 A 获知有 4 条等价路径通往路由器 B，每条路径都要经过一个节点。ECMP 用于在这 4 条路径之间对流量执行负载均衡。每个节点在与外部路由器通信时，都会挑选不同的路由器 ID。

您必须为路由器 ID 配置一个集群池，使每个节点都有单独的路由器 ID。

EIGRP 与单个接口模式下的集群对等体不构成邻居关系。

**注释** 如果该集群为实现冗余有多个设备与同一个路由器相邻，则非对称路由可能会造成不可接受的流量损失。要避免非对称路由，请将所有这些节点接口分组到同一流量区域中。请参阅配置流量区域。

## FTP 和集群

- 如果 FTP 数据通道和控制通道流量由不同的集群成员所有，则数据通道所有者会将空闲超时更新定期发送到控制通道所有者并更新空闲超时值。但是，如果重新加载控制流量所有者并重新托管控制流量，则不会再保持父/子流量关系；控制流量空闲超时不会更新。
- 如果您将 AAA 用于 FTP 访问，则控制通道流集中在控制节点上。

## ICMP检测和集群

ICMP 和 ICMP 错误数据包通过集群的流取决于是否启用 ICMP/ICMP 错误检查。不启用 ICMP 检查时，ICMP 是单向流，并且不支持导向器流。启用 ICMP 检查时，ICMP 流变为双向流，并由导向器/备份流支持。被检查的 ICMP 流的一个不同之处在于导向器对转发数据包的处理：导向器会将 ICMP 回应应答数据包转发给流所有者，而不是将数据包返回给转发器。

## 组播路由和集群

在独立接口模式下，设备并不单独处理组播。所有数据和路由数据包都由控制设备进行处理和转发，从而避免数据包复制。

## NAT 和集群

NAT 可能会影响集群的总吞吐量。入站和出站 NAT 数据包可被发送到集群中不同的 ASA，因为负载均衡算法取决于 IP 地址和端口，NAT 会导致入站和出站数据包具有不同的 IP 地址和/或端口。当数据包到达并非 NAT 所有者的 ASA时，会通过集群控制链路转发到所有者，导致集群控制链路上存在大量流量。请注意，接收节点不会创建流向所有者的转发流量，因为 NAT 所有者最终可能不会根据安全和策略检查结果为数据包创建连接。

如果您仍想在集群中使用 NAT，请考虑以下准则：

- 不使用代理 ARP - 对于独立接口，切勿为映射的地址发送代理 ARP 回复。这可以防止邻接路由器与可能已经不在集群中的 ASA 保持对等关系。对于指向主集群 IP 地址的映射地址，上游路由器需要静态路由或带对象跟踪的 PBR。这对跨网络 EtherChannel 来说不是问题，因为只有一个 IP 地址与集群接口关联。

- PAT 采用端口块分配 - 请参阅该功能的以下准则：
  - 每主机最大流量限制并不针对整个集群，而是单独应用于每个节点。因此，在每主机最大流量限制配置为 1 的包含 3 个节点的集群中，如果在全部 3 个节点上对来自主机的流量实行负载均衡，则可以分配 3 个端口块，每个节点 1 个。
  - 在执行每主机最大流量限制时，在备份池中的备份节点上创建的端口块不计算在内。
  - 如果进行即时 PAT 规则修改（对 PAT 池改用全新的 IP 地址范围），会导致在新的池生效时仍在传输的转换项备份请求的转换项备份创建失败。此行为并非端口块分配功能所特有，它是一个暂时性 PAT 池问题，只发现于在集群节点之间分配池并执行流量负载均衡的集群部署。
  - 在集群中操作时，不能直接更改块分配大小。只有在集群中重新加载每个设备后，新的大小才会生效。为避免重新加载每个设备，我们建议您删除所有块分配规则并清除与这些规则相关的所有转换。然后，您可以更改块大小并重新创建块分配规则。

- 对动态 PAT 使用 NAT 池地址分配 - 配置 PAT 池时，集群将池中的每个 IP 地址划分为端口块。默认情况下，每个块都是 512 个端口，但如果配置端口块分配规则，则使用块设置。这些块在集群中的各个节点之间均匀分配，因此每个节点都有一个或多个块对应 PAT 池中的每个 IP 地址。因此，在一个集群的 PAT 池中可以最少拥有一个 IP 地址，只要这足以支持您预期的 PAT

连接数即可。端口块覆盖的端口范围为 1024-65535，除非您在 PAT 池 NAT 规则中配置该选项以包含保留的端口 1-1023。

- 在多个规则中重复使用 PAT 池 - 要在多条规则中使用同一 PAT 池，必须注意规则中的接口选择。必须在所有规则中使用特定接口，或者在所有规则中使用“任意”接口。不能在规则中混合使用特定接口和“任意”接口，否则系统可能无法将返回流量与集群中的正确节点进行匹配。每条规则使用唯一的 PAT 池是最可靠的方案。
- 不使用轮询 - 集群不支持 PAT 池轮询。
- 无扩展 PAT - 集群不支持扩展 PAT。
- 控制节点管理的动态 NAT 转换 - 控制节点保留转换表并复制到数据节点。当数据节点收到需要动态 NAT 的连接并且转换不在表中时，它将请求从控制节点转换。数据节点拥有该连接。
- 过时 xlate - 连接所有者上的 xlate 空闲时间不会更新。因此，空闲时间值可能会超过空闲超时值。如果空闲计时器值高于配置的超时值（refcnt 为 0），则表示 xlate 过时。
- 每会话 PAT 功能 - 每会话 PAT 功能并非集群专用功能，但它能提高 PAT 的可扩展性，而且对集群而言，它允许每个数据节点成为 PAT 连接的所有者；相比之下，多会话 PAT 连接则必须转发到控制节点并由控制节点所有。默认情况下，所有 TCP 流量和 UDP DNS 流量均使用每会话 PAT 转换，而 ICMP 和所有其他 UDP 流量均使用多会话。您可以为 TCP 和 UDP 配置每会话 NAT 规则以更改这些默认设置，但是，您不能为 ICMP 配置每会话 PAT。例如，与通过 TCP/443 的 HTTPS TLS 相比，通过 UDP/443 的 Quic 协议是性能最佳的替代方案，随着它的使用越来越多，应该为 UDP/443 启用每个会话 PAT。对于使用多会话 PAT 的流量（例如 H.323、SIP 或 Skinny），您可以禁用关联 TCP 端口的每会话 PAT（这些 H.323 和 SIP 的 UDP 端口已默认为多会话）。有关每会话 PAT 的详细信息，请参阅防火墙配置指南。
- 对以下检查不使用静态 PAT：
  - FTP
  - PPTP
  - RSH
  - SQLNET
  - TFTP
  - XDMCP
  - SIP
- 如果您有大量 NAT 规则（超过一万条），则应使用设备 CLI 中的 **asp rule-engine transactional-commit nat** 命令启用事务提交模型。否则，节点可能无法加入集群。

## SCTP 和集群

SCTP 关联可以在任何节点上创建（由于负载均衡），但其多宿主连接必须位于同一节点上。

## SIP 检测和集群

控制流可以在任何节点上创建（由于负载均衡），但其子数据流必须位于同一节点上。

不支持 TLS 代理配置。

## SNMP 和集群

SNMP 代理按照本地 IP 地址轮询每一个ASA。您无法轮询集群的合并数据。

您应始终使用本地地址而非主集群 IP 地址进行 SNMP 轮询。如果 SNMP 代理轮询主集群 IP 地址，则当选举出新的控制节点时，对新控制节点的轮询将失败。

当使用 SNMPv3 进行集群时，如果您在初始集群形成后添加新的集群节点，则 SNMPv3 用户不会复制到新节点。您必须在控制节点上重新添加它们以强制用户复制到新节点，或者直接在数据节点上复制。

## STUN 和集群

故障转移和集群模式支持STUN检查，因为针孔被复制。但是，节点之间不进行事务ID的复制。如果节点在收到 STUN 请求后发生故障，并且另一个节点收到 STUN 响应，则该 STUN 响应将被丢弃。

## 系统日志和集群

- 系统日志 - 集群中的每个节点都会生成自己的系统日志消息。您可以配置日志记录，使每个节点在系统日志消息的报头字段中使用相同或不同的设备 ID。例如，集群中的所有节点都会复制和共享主机名配置。如果将日志记录配置为使用主机名作为设备 ID，则所有节点生成的系统日志消息都会看似来自一个节点。如果将日志记录配置为使用集群引导程序配置中指定的本地节点名称作为设备 ID，系统日志消息就会看似来自不同节点。
- NetFlow - 集群中的每个节点都会生成自己的 NetFlow 数据流。NetFlow 收集器只能将每台 ASA 视为单独的 NetFlow 导出器。

## 思科 TrustSec 和集群

只有控制节点学习安全组标记 (SGT) 信息。然后，控制节点将 SGT 填充到数据节点，数据节点可以根据安全策略对 SGT 做出匹配决策。

## VPN 和集群

站点间 VPN 是集中功能；只有控制节点支持 VPN 连接。

注释 集群不支持远程访问 VPN。

VPN 功能仅限控制节点使用，且不能利用集群的高可用性功能。如果控制节点发生故障，所有现有的 VPN 连接都将断开，VPN 用户将遇到服务中断。选择新的控制节点后，必须重新建立 VPN 连接。

对于使用 PBR 或 ECMP 时与独立接口的连接，您必须始终连接到主集群 IP 地址而非本地地址。

与 VPN 相关的密钥和证书将被复制到所有节点。

# 性能换算系数

将多台设备组成一个集群时，预计可以达到近似 80% 最大组合吞吐量的集群总体性能。

例如，如果您的型号在单独运行时可以处理大约 10 Gbps 的流量，则对于 8 台设备的集群，最大组合吞吐量约为 80 Gbps（8 台设备 x 10 Gbps）的 80%：64 Gbps。

# 控制节点选择

集群节点通过集群控制链路通信，如下选举控制节点：

1. 当为节点启用集群（或当节点首次启动时已启用集群）时，设备会每 3 秒广播一个选举请求。
2. 具有较高优先级的任何其他设备都会响应选举请求；优先级设置在 1 和 100 之间，其中 1 为最高优先级。
3. 如果某节点在 45 秒后未收到另一个具有较高优先级的节点的响应，则该设备会成为控制节点。

**注释** 如果多个节点并列获得最高优先级，则使用集群节点名称和序列号确定控制节点。

4. 如果节点稍后加入具有更高优先级的集群，它不会自动成为控制节点；现有控制节点始终保持为控制节点，除非它停止响应，此时会选择新的控制节点。
5. 在“裂脑”场景中，当临时存在多个控制节点时，具有最高优先级的节点将会保留角色，而其他节点则恢复为数据节点角色。

**注释** 您可以手动强制节点成为控制节点。对集中功能而言，如果强制更改控制节点，则所有连接都将断开，而您必须在新的控制节点上重新建立连接。

# 集群中的高可用性

集群通过监控节点和接口的运行状况并在节点之间复制连接状态来提供高可用性。

## 节点运行状况监控

每个节点通过集群控制链路定期发送广播保持连接心跳数据包。如果控制节点在可配置的超时期限内未从数据节点接收任何keepalive心跳数据包或其他数据包，则控制节点会从集群中删除该数据节点。如果数据节点未从控制节点接收数据包，则从其余节点中选择新的控制节点。

如果节点因为网络故障而不是因为节点实际故障而无法通过集群控制链路相互访问，则集群可能会进入“裂脑”场景，其中隔离的数据节点将选择自己的控制节点。例如，如果路由器在两个集群位置之间发生故障，则位于位置1的原始控制节点将从集群中删除位置2数据节点。同时，位置2的节点将选择自己的控制节点并形成自己的集群。请注意，在这种情况下，非对称流量可能会失败。恢复集群控制链路后，优先级较高的控制节点将保留控制节点的角色。

## 接口监控

每个节点都会监控使用中的所有已命名的硬件接口的链路状态，并向控制节点报告状态更改。

当您启用运行状况监控时，默认情况下会监控所有物理接口；您可以选择按接口禁用监控。只能监控已命名接口。

如果某个节点被监控的接口发生故障，则将从集群中删除该设备。ASA 在多长时间后从集群中删除成员取决于该节点是既定成员还是正在加入集群的设备。ASA 在节点加入集群后的最初 90 秒不监控接口。在此期间的接口状态更改不会导致 ASA 从集群中删除。无论状态如何，节点都会在 500 毫秒后被删除。

## 发生故障后的状态

如果控制节点发生故障，则优先级最高（数字最小）的另一个集群成员将成为控制节点。

ASA将自动尝试重新加入集群，具体取决于故障事件。

**注释** 当ASA变成不活动状态且无法自动重新加入集群时，所有数据接口都会关闭，仅管理专用接口可以发送和接收流量。管理接口将保持打开，使用节点从集群 IP 池接收的 IP 地址。但是，如果您重新加载而节点在集群中仍然处于非活动状态，管理接口将被禁用。您必须使用控制台端口来进行任何进一步配置。

## 重新加入集群

当集群节点从集群中删除之后，如何才能重新加入集群取决于其被删除的原因：

- 集群控制链路在最初加入时出现故障 - 在解决集群控制链路存在的问题后，您必须通过在 CLI 输入 **cluster group***name*，然后输入 **enable** 来重新启用集群 ，以手动重新加入集群。
- 加入集群后出现故障的集群控制链路 - ASA 无限期地每 5 分钟自动尝试重新加入。此行为是可配置的。
- 出现故障的数据接口 - ASA 尝试在 5 分钟时、然后在 10 分钟时、最后在 20 分钟时重新加入。如果 20 分钟后仍加入失败，ASA 将禁用集群。解决数据接口问题之后，您必须在 CLI 上通过输入**cluster group** *name*，然后输入**enable** 来手动启用集群。此行为是可配置的。

- 节点存在故障 - 如果节点因节点运行状况检查失败而从集群中删除，则如何重新加入集群取决于失败的原因。例如，临时电源故障意味节点将在重新启动时重新加入集群，只要集群控制链路打开并且仍然使用 **enable** 命令启用集群。ASA 每 5 秒钟尝试重新加入集群。
- 内部错误 - 内部故障包括：应用同步超时；应用状态不一致等。节点将尝试以下列间隔自动重新加入集群：5 分钟，10 分钟，然后是 20 分钟。此行为是可配置的。

## 数据路径连接状态复制

每个连接在集群中都有一个所有者和至少一个备用所有者。备用所有者在发生故障时不会接管连接；而是存储 TCP/UDP 状态信息，使连接在发生故障时可以无缝转移到新的所有者。备用所有者通常也是导向器。

有些流量需要 TCP 或 UDP 层以上的状态信息。请参阅下表了解支持或不支持此类流量的集群。

**表 *3:* 在集群中复制的功能**

| 流量 | 状态支持 | 备注 |
|---|---|---|
| 运行时间 | 是 | 跟踪系统运行时间。 |
| ARP 表 | 是 | - |
| MAC 地址表 | 是 | - |
| 用户标识 | 是 | 包括 AAA 规则 (uauth) 。 |
| IPv6 邻居数据库 | 是 | — |
| 动态路由 | 是 | — |
| SNMP 引擎 ID | 否 | - |
| 适用于 Firepower 4100/9300的分布式 VPN（站点间） | 是 | 备用会话成为主用会话，并创建一个新的备用会话。 |

# 集群管理连接的方式

连接可以负载平衡到集群的多个节点。连接角色决定了在正常操作中和高可用性情况下处理连接的方式。

## 连接角色

请参阅为每个连接定义的下列角色：

- 所有者 - 通常为最初接收连接的节点。所有者负责维护 TCP 状态并处理数据包。一个连接只有一个所有者。如果原始所有者发生故障，则当新节点从连接接收到数据包时，导向器会从这些节点中选择新的所有者。

- 备用所有者 - 存储从所有者接收的 TCP/UDP 状态信息的节点，以便在出现故障时可以无缝地将连接转移到新的所有者。在发生故障时，备用所有者不会接管连接。如果所有者处于不可用状态，从该连接接收数据包的第一个节点（根据负载均衡而定）联系备用所有者获取相关的状态信息，以便成为新的所有者。

  只要导向器（见下文）与所有者不是同一节点，导向器也可以是备用所有者。如果所有者选择自己作为导向器，则选择一个单独的备用所有者。

  对于 Firepower 9300 上的在一个机箱中包括多达 3 个集群节点的集群，如果备用所有者与所有者在同一机箱上，则将从另一个机箱中选择一个额外的备用所有者来保护流量免受机箱故障的影响。

  如果您对站点间集群启用导向器本地化，则有两个备用所有者角色：本地备用和全局备用。所有者始终选择与自身位于同一站点（基于站点 ID）的本地备用。全局备用可以位于任何站点，甚至可以与本地备用是同一个节点。所有者向两个备用所有者发送连接状态信息。

  如果启用站点冗余，并且备用所有者与所有者位于同一站点，则将从另一个站点选择一个额外的备用所有者来保护流量免受站点故障的影响。机箱备份和站点备份是独立的，因此流量在某些情况将同时具有机箱备份和站点备份。

- 导向器 - 处理来自转发器的所有者查找请求的节点。当所有者收到新连接时，会根据源/目的 IP 地址和端口的散列值（有关 ICMP 散列详细信息，请参见下文）选择导向器，然后向导向器发送消息来注册该新连接。如果数据包到达除所有者以外的任何其他节点，该节点会向导向器查询哪一个节点是所有者，以便转发数据包。一个连接只有一个导向器。如果导向器发生故障，所有者会选择一个新的导向器。

  只要导向器与所有者不是同一节点，导向器也可以是备用所有者（见上文）。如果所有者选择自己作为导向器，则选择一个单独的备用所有者。

  如果您对站点间集群启用导向器本地化，则有两个导向器角色：本地导向器和全局导向器。所有者始终选择与它自身位于同一站点（基于站点 ID）的本地导向器。全局导向器可以位于任何站点，甚至可以与本地导向器是同一节点。如果原始所有者发生故障，本地导向器将在同一站点选择新的连接所有者。

  ICMP/ICMPv6 散列详细信息：

  - 对于 Echo 数据包，源端口为 ICMP 标识符，目的端口为 0。
  - 对于 Reply 数据包，源端口为 0，目的端口为 ICMP 标识符。
  - 对于其他数据包，源端口和目的端口均为 0。

- 转发器 - 向所有者转发数据包的节点。如果转发者收到并非其所有的连接的数据包，则会向导向器查询所有者，然后为其收到的此连接的任何其他数据包建立发往所有者的流量。导向器也可以是转发者。如果启用导向器本地化，转发器将始终向本地导向器查询。如果本地导向器未获知所有者，例如，当集群成员收到所有者位于其他站点上的连接的数据包时，转发者将仅查询全局导向器。请注意，如果转发者收到 SYN-ACK 数据包，它可以从数据包的 SYN Cookie 直接获知所有者，因此无需向导向器查询。（如果禁用 TCP 序列随机化，则不会使用 SYN Cookie；必须向导向器查询。）对于 DNS 和 ICMP 等持续时间极短的流量，转发者不会查询，而是立即将数据包发送到导向器，然后由其发送到所有者。一个连接了可以有多个转发器；采用良好的

负载均衡方法可以做到没有转发器，让一个连接的所有数据包都由所有者接收，从而实现最高效率的吞吐量。

注释 不建议您在使用集群时禁用 TCP 序列随机化。由于 SYN/ACK 数据包可能会被丢弃，因此少数情况下可能无法建立某些 TCP 会话。

- 分段所有者 - 对于分段的数据包，接收分段的集群节点使用分段源 IP 地址、目的 IP 地址和数据包 ID 的散列确定分段所有者。然后，所有片段都通过集群控制链路转发给片段所有者。片段可能均衡分发给不同的集群节点，因为只有第一个分段包含交换机负载均衡散列中使用的 5 元组。其他片段不包含源端口和目的端口，可能会均衡分发给其他集群节点。片段所有者临时重组数据包，以便根据源/目标 IP 地址和端口的散列来确定导向器。如果是新连接，则片段所有者将注册为连接所有者。如果是现有连接，则片段所有者将所有片段转发给集群控制链路上提供的连接所有者。然后，连接所有者将重组所有片段。

当连接使用端口地址转换 (PAT) 时，PAT 类型（每会话或多会话）会对哪个集群成员将成为新连接的所有者产生影响：

- 每会话 PAT - 所有者是接收连接中的初始数据包的节点。

  默认情况下，TCP 和 DNS UDP 流量均使用每会话 PAT。

- 多会话 PAT - 所有者始终是控制节点。如果多会话 PAT 连接最初由数据节点接收，则数据节点会将连接转发给控制节点。

  默认情况下，UDP（DNS UDP 除外）和 ICMP 流量使用多会话 PAT，因此这些连接始终归控制节点所有。

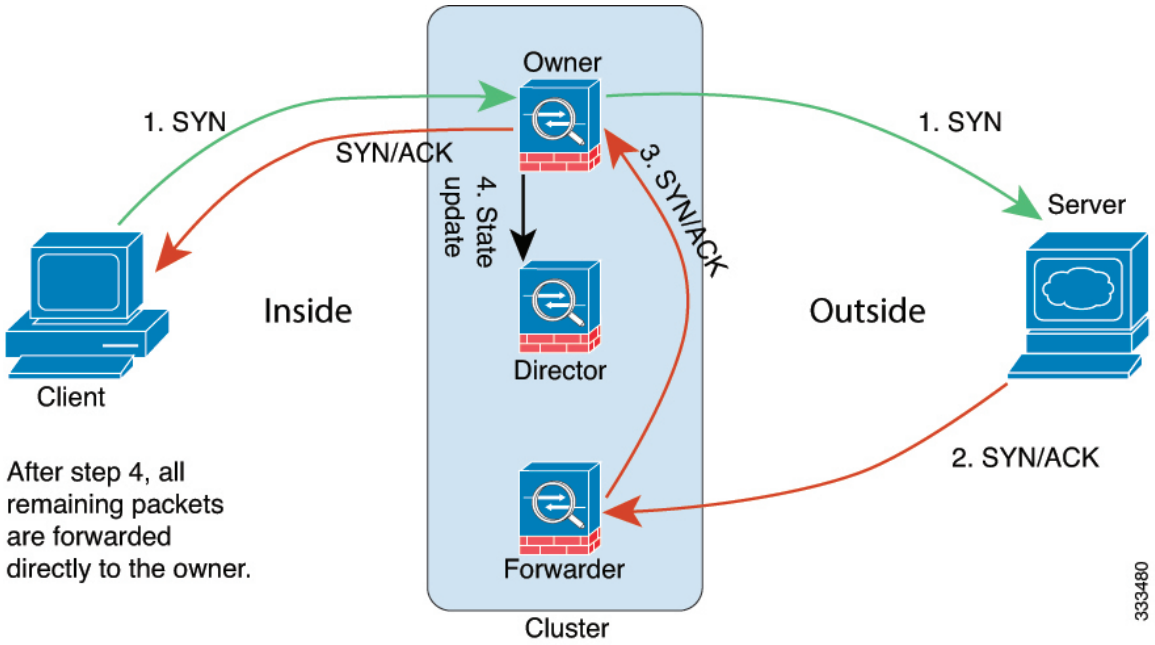
您可以更改 TCP 和 UDP 的每会话 PAT 默认设置，以便根据配置按每会话或多会话处理这些协议的连接。对于 ICMP，您不能更改默认的多会话 PAT。有关每会话 PAT 的详细信息，请参阅防火墙配置指南。

## 新连接所有权

通过负载均衡将新连接定向到集群节点时，该连接的两个方向都由此节点所有。如果该连接有任何数据包到达其他节点，这些数据包都会通过集群控制链路被转发到所有者节点。为了获得最佳性能，对于要到达同一个节点的流量的两个方向以及要在节点之间均摊的流量，都需要执行适当的外部负载均衡。如果反向流量到达其他节点，会被重定向回原始节点。

## TCP 的数据流示例

以下图例显示了新连接的建立。

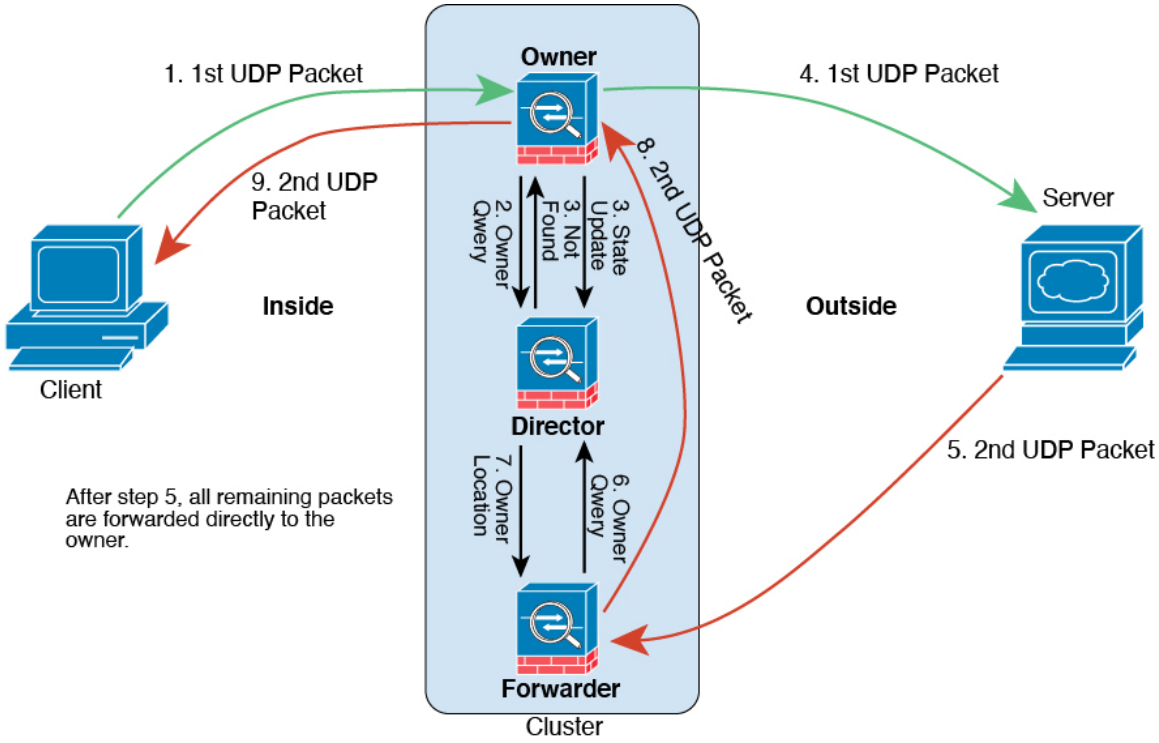



1. SYN 数据包从客户端发出，被传送到一台 ASA（基于负载均衡方法），该设备将成为所有者。所有者创建一个流量，将所有者信息编码为 SYN Cookie，然后将数据包转发到服务器。
2. SYN-ACK 数据包从服务器发出，被传送到一台不同的 ASA（基于负载均衡方法）。此 ASA是转发者。
3. 由于转发器不是该连接的所有者，因此它将解码 SYN Cookie 中的所有者信息，然后创建发往所有者的转发流量，并将 SYN-ACK 数据包转发到所有者。
4. 所有者将状态更新发送到导向器，然后将 SYN-ACK 数据包转发到客户端。
5. 导向器接收来自所有者的状态更新，创建发往所有者的流量，并记录TCP状态信息以及所有者。导向器将充当该连接的备用所有者。
6. 传送到转发器的任何后续数据包都会被转发到所有者。
7. 如果数据包被传送到任何其他节点，它将向导向器查询所有者并建立一个流量。
8. 该流量的任何状态更改都会导致所有者向导向器发送状态更新。

## ICMP 和 UDP 的数据流示例

以下图例显示了新连接的建立。

1. 图 *13: ICMP* 和 *UDP* 数据流




第一个 UDP 数据包从客户端发出，被传送到一个ASA（基于负载均衡方法）。

2. 收到第一个数据包的节点查询基于源/目的 IP 地址和端口的散列值选择的导向器节点。
3. 导向器找不到现有流，创建导向器流并将数据包转发回前一个节点。换句话说，导向器已为此流选择了所有者。
4. 所有者创建流，向导向器发送状态更新，然后将数据包转发到服务器。
5. 第二个 UDP 数据包从服务器发出，并被传送到转发器。
6. 转发器向导向器查询所有权信息。对于 DNS 等持续时间极短的流量，转发者不会查询，而是立即将数据包发送到导向器，然后由其发送到所有者。
7. 导向器向转发器回复所有权信息。
8. 转发器创建转发流以记录所有者信息，并将数据包转发给所有者。
9. 所有者将数据包转发到客户端。

# 关于公共云中的 ASA Virtual 集群的历史记录

| 特性 | Version | 详细信息 |
|---|---|---|
| 关于节点加入的 MTU ping 测试 | 9.23(1) | 当某个节点加入集群时，它会向控制节点发送 ping，其数据包大小与集群控制链路 MTU 匹配，从而检查 MTU 兼容性。如果 ping 失败，系统会生成通知，以便您纠正连接的交换机上 MTU 不匹配的问题，然后重试。 |
| Azure 上的 ASA Virtual：使用网关负载均衡的集群 | 9.20(2) | 我们现在支持使用 Azure 资源管理器 (ARM) 模板在 Azure 上部署 ASA Virtual 集群，然后将 ASA Virtual 配置为使用网关负载均衡器 (GWLB) 来实现网络流量负载均衡。 |
| 在 AWS 中使用 GWLB 配置 ASA Virtual 集群的目标故障转移 | 9.20(2) | AWS 中的目标故障转移功能使 GWLB 能够在计划维护期间取消注册或目标节点发生故障的情况下将网络数据包重定向到正常运行的目标节点。它利用集群的状态故障切换。 |
| 流状态的可配置集群保持连接间隔 | 9.20(2) | 流所有者向导向器和备份所有者发送保持连接（clu_keepalive 消息）和更新（clu_update 消息），以刷新流状态。您现在可以设置保持连接的间隔。默认值为 15 秒，您也可以将间隔设为 15 到 55 秒。您可能想将间隔设置得更长，以减少集群控制链路上的流量。<br>新增/修改的命令：**clu-keepalive-interval** |
| ASA Virtual Amazon Web Services (AWS) 集群 | 9.19(1) | ASA Virtual 支持 AWS 上最多 16 个节点的单个接口集群。无论是否有 AWS 网关负载均衡器，您都可以使用集群。 |
| Amazon Web 服务 (AWS) 集群中的 ASA Virtual IMDSv2 支持 | 9.22 | ASA Virtual 支持 AWS 上的 IMDSv2。您可以通过更新堆栈来启用“IMDSv2 必需”模式。 |

当地语言翻译版本说明

思科可能会在某些地方提供本内容的当地语言翻译版本。请注意，翻译版本仅供参考，如有任何不一致之处，以本内容的英文版本为准。