



## 在 OpenStack 上部署 ASAv

您可以在 OpenStack 上部署 ASAv。

- [关于在 OpenStack 上的 ASAv 部署，第 1 页](#)
- [ASAv 和 OpenStack 的前提条件，第 1 页](#)
- [ASAv 和 OpenStack 的准则和限制，第 2 页](#)
- [OpenStack 要求，第 3 页](#)
- [OpenStack 上的 ASAv 网络拓扑示例，第 4 页](#)
- [在 OpenStack 上部署 ASAv，第 5 页](#)

## 关于在 OpenStack 上的 ASAv 部署

您可以在 OpenStack 环境中部署思科自适应安全虚拟设备 (ASAv)。OpenStack 是一套用于构建和管理适用于公共云和私有云的云计算平台的软件工具，并且与 KVM 虚拟机监控程序紧密集成。

通过为 ASAv 启用 OpenStack 平台支持，您可以在开源云平台上运行思科 ASAv。OpenStack 使用 KVM 虚拟机监控程序来管理虚拟资源。KVM 虚拟机监控程序已支持 ASAv 设备。因此，无需额外添加内核软件包或驱动程序即可启用 OpenStack 支持。

## ASAv 和 OpenStack 的前提条件

- 从 [software.cisco.com](http://software.cisco.com) 下载 ASAv qcow2 文件并将其放在 Linux 主机上：  
<http://www.cisco.com/go/asa-software>
- ASAv 支持在开源 OpenStack 环境和思科 VIM 托管 OpenStack 环境中进行部署。

根据 OpenStack 指南来设置 OpenStack 环境。

- 请参阅开源 OpenStack 文档：<https://docs.openstack.org/project-deploy-guide/openstack-ansible/stein/overview.html>
- 请参阅思科虚拟化基础设施管理器 (VIM) OpenStack 文档：[思科虚拟化基础设施管理器文档，3.4.3 至 3.4.5](#)

- 许可 ASAv。在您许可 ASAv 之前，ASAv 将在降级模式下运行，此模式仅支持 100 个连接和 100 Kbps 的吞吐量。请参阅[许可证：智能软件许可（Firepower 上的 ASAv、ASA）](#)。
- 接口要求：
  - 管理接口
  - 内部和外部接口
- 通信路径：
  - 管理接口 - 用于将 ASAv 连接到 ASDM；不能用于流量。
  - 内部接口（必需） - 用于将 ASAv 连接到内部主机。
  - 外部接口（必需） - 用于将 ASAv 连接到公共网络。
- 通信路径：
  - 用于访问 ASAv 的浮动 IP。
- 最低支持的 ASAv 版本：
  - ASA 9.16.1
- 有关 OpenStack 要求，请参阅[OpenStack 要求](#)。
- 有关 ASAv 的系统要求，请参阅[思科 ASA 兼容性矩阵](#)。

## ASAv 和 OpenStack 的准则和限制

### 支持的功能

OpenStack 上的 ASAv 支持以下功能：

- 在 OpenStack 环境中在计算节点上运行的 KVM 虚拟机监控程序上部署 ASAv。
- OpenStack CLI
- 基于 Heat 模板的部署
- OpenStack Horizon 控制面板
- 路由模式（默认）
- 许可 - 仅支持 BYOL
- 使用 CLI 和 ASDM 管理 ASAv
- 驱动程序 - VIRTIO、VPP 和 SRIOV

### 不支持的功能

OpenStack 上的 ASAv 不支持以下各项：

- Autoscale
- OpenStack 版本，而不是 OpenStack Stein 和 Queens 版本
- Ubuntu 18.04 版本和 Red Hat Enterprise Linux (RHEL) 7.6 之外的操作系统

## OpenStack 要求

OpenStack 环境必须符合以下支持的硬件和软件要求。

表 1: 硬件和软件要求

类别	支持的版本	说明
服务器	UCS C240 M5	建议使用 2 台 UCS 服务器，分别用于 os-controller 和 os-compute 节点。
驱动程序	VIRTIO、IXGBE、I40E	这些是支持的驱动程序。
操作系统	Ubuntu Server 18.04	这是 UCS 服务器上的建议操作系统。
OpenStack 版本	Stein 版本	有关各种 OpenStack 版本的详细信息，请访问： <a href="https://releases.openstack.org/">https://releases.openstack.org/</a>

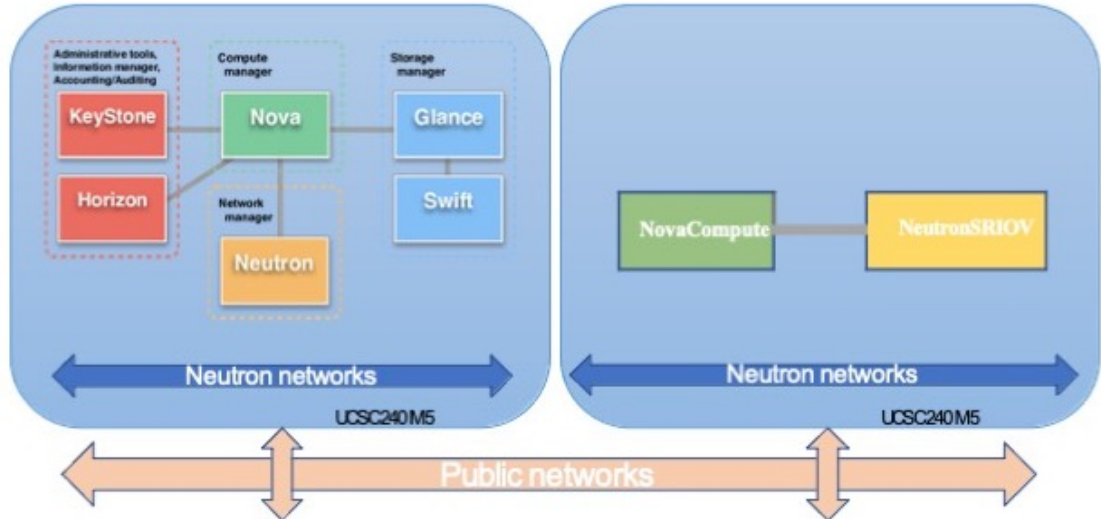
表 2: 思科 VIM 托管 OpenStack 的硬件和软件要求

类别	支持的版本	说明
服务器硬件	UCS C220-M5/UCS C240-M4	建议使用 5 台 UCS 服务器，其中 3 台用于 os-controller，两台或更多用于 os-compute 节点。
驱动因素	VIRTIO、SRIOV 和 VPP	这些是支持的驱动程序。
操作系统	Red Hat Enterprise Linux 7.6	这是建议的操作系统。
OpenStack 版本	OpenStack 13.0 (Queens 版本)	有关各种 OpenStack 版本的详细信息，请访问： <a href="https://releases.openstack.org/">https://releases.openstack.org/</a>

类别	支持的版本	说明
思科 VIM 版本	思科 VIM 3.4.4	请参阅 <a href="#">思科 VIM OpenStack 文档</a> 。

图 1: OpenStack 平台拓扑

OpenStack 平台拓扑会显示两台 UCS 服务器上的常规 OpenStack 设置。



## OpenStack 上的 ASA 网络拓扑示例

下图显示了在路由防火墙模式下建议用于 ASA 的网络拓扑，在 OpenStack 中为 ASA 配置了 3 个子网（管理、内部和外部）。

图 2: OpenStack 部署上的 AWS 示例



## 在 OpenStack 上部署 ASAv

思科提供用于部署 ASAv 的示例 Heat 模板。创建 OpenStack 基础设施资源的步骤汇总在 Heat 热模板 (Deploy\_os\_infra.yaml) 文件中，以创建网络、子网和路由器接口。总体而言，ASAv 部署步骤分为以下几个部分。

- 将 ASAv qcow2 映像上传到 OpenStack Glance 服务。
- 创建网络基础设施。
  - 网络
  - 子网
  - 路由器接口
- 创建 ASAv 实例。
  - 类型
  - 安全组
  - 浮动 IP
  - 实例

您可以按照以下步骤在 OpenStack 上部署 ASAv。

### 将 ASAv 映像上传到 OpenStack

将 qcow2 映像 (asav-<version>.qcow2) 复制到 OpenStack 控制器节点，然后将映像上传到 OpenStack Glance 服务。

开始之前

从 Cisco.com 下载 ASAv qcow2 文件并将其放在 Linux 主机上：

<http://www.cisco.com/go/asa-software>



---

注释 需要 Cisco.com 登录信息和思科服务合同。

---

**步骤 1** 将 qcow2 映像文件复制到 OpenStack 控制器节点。

**步骤 2** 将 ASAv 映像上传到 OpenStack Glance 服务。

```
root@ucs-os-controller:~$ openstack image create <image_name> --public --disk-format qcow2 --container-format bare --file ./<asav_qcow2_file>
```

步骤 3 验证 ASAv 映像上传是否成功。

```
root@ucs-os-controller:$ openstack image list
```

示例:

```
root@ucs-os-controller:$ openstack image
list+-----+
| ID                                     | Name                               | Status |
|+-----+-----+-----+
| 06dd7975-0b6e-45b8-810a-4ff98546a39d | asav-<version>-image             | active |
|+-----+-----+-----+
```

系统将显示已上传的映像及其状态。

下一步做什么

使用 `deploy_os_infra.yaml` 模板来创建网络基础设施。

## 为 OpenStack 和 ASAv 创建网络基础设施

开始之前

需要使用 Heat 模板文件来创建网络基础设施和 ASAv 所需的组件，例如终端、网络、子网、路由器接口和安全组规则：

- `deploy_os_infra.yaml`
- `env.yaml`

您的 ASAv 版本的模板可从 GitHub 存储库获取：

- <https://github.com/CiscoDevNet/cisco-asav>



### 重要事项

请注意，思科提供的模板作为开源示例提供，不在常规思科 TAC 支持范围内。定期检查 GitHub 以了解更新和自述文件说明。

步骤 1 部署基础设施 Heat 模板文件。

```
root@ucs-os-controller:$ openstack stack create <stack-name> -e <environment files name> -t <deployment file name>
```

示例:

```
root@ucs-os-controller:$ openstack stack create infra-stack -e env.yaml -t deploy_os_infra.yaml
```

步骤 2 验证是否已成功创建基础设施堆栈。

```
root@ucs-os-controller:$ openstack stack list
```

下一步做什么

在 OpenStack 上创建 ASAv 实例。

## 在 OpenStack 上创建 ASAv 实例

使用示例 Heat 模板在 OpenStack 上部署 ASAv。

开始之前

在 OpenStack 上部署 ASAv 需要 Heat 模板：

- `deploy_asav.yaml`

您的 ASAv 版本的模板可从 GitHub 存储库获取：

- <https://github.com/CiscoDevNet/cisco-asav>



### 重要事项

请注意，思科提供的模板作为开源示例提供，不在常规思科 TAC 支持范围内。定期检查 GitHub 以了解更新和自述文件说明。

**步骤 1** 部署 ASAv 热模板文件 (`deploy_asav.yaml`) 以创建 ASAv 实例。

```
root@ucs-os-controller:~$ openstack stack create asav-stack -e env.yaml -t deploy_asav.yaml
```

示例：

```
+-----+-----+
| Field          | Value                                     |
+-----+-----+
| id             | 14624af1-e5fa-4096-bd86-c453bc2928ae |
| stack_name     | asav-stack                             |
| description    | ASAvtemplate                           |
| creation_time  | 2020-12-07T14:55:05Z                   |
| updated_time   | None                                     |
| stack_status   | CREATE_IN_PROGRESS                     |
| stack_status_reason | Stack CREATE started                   |
+-----+-----+
```

**步骤 2** 验证是否已成功创建 ASAv 堆栈。

```
root@ucs-os-controller:~$ openstack stack list
```

示例：

```
+-----+-----+-----+-----+-----+-----+
| ID                | Creation Time | Updated Time | Stack Name | Project | Stack Status |
+-----+-----+-----+-----+-----+-----+
| 14624af1-e5fa-4096-bd86-c453bc2928ae | 2020-12-07T14:55:05Z | None | asav-stack | 13206e49b48740fdafca83796c6f4ad5 | CREATE_COMPLETE |
| 198336cb-1186-45ab-858f-15ccd3b909c8 |  |  | infra-stack | 13206e49b48740fdafca83796c6f4ad5 | CREATE_COMPLETE |
+-----+-----+-----+-----+-----+-----+
```

| 2020-12-03T10:46:50Z | None |

---

---