



在工作负载上部署软件代理

Cisco Secure Workload 软件代理是在工作负载上安装的轻量级软件。代理的目的是：

- 收集主机信息，如网络接口和系统中运行的活动进程。
- 监控和收集网络流信息。
- 通过为安装并启用了软件代理的主机设置防火墙规则来执行安全策略。

当接口地址更改时，代理会自动更新 Cisco Secure Workload 资产。您无需在最终用户（员工）计算机上安装代理。

- [部署软件代理, on page 2](#)
- [安全排除项, on page 27](#)
- [代理的服务管理, on page 30](#)
- [使用代理的执行策略, on page 32](#)
- [软件代理配置, 第 56 页](#)
- [在工作负载配置文件中查看详细的代理状态, on page 66](#)
- [重新连接代理, on page 68](#)
- [生成代理令牌, on page 71](#)
- [启用执行时主机 IP 地址更改, on page 72](#)
- [升级软件代理, 第 73 页](#)
- [删除软件代理, 第 77 页](#)
- [工作负载代理收集和导出的数据, on page 80](#)
- [执行警报, on page 83](#)
- [传感器警报, on page 89](#)
- [常见问题解答, on page 95](#)

部署软件代理



Note 注销后，使用自动角色映射从 LDAP 或 AD 帐户下载安装程序脚本会失败。要让安装程序脚本不间断地访问集群，请启用“使用本地身份验证”(Use Local Authentication)。

在部署时，Cisco Secure Workload 集群会根据运行代理的主机特定的一组参数来为代理分配唯一身份。如果主机名和 BIOS UUID 是参数集的一部分，则可能会遇到以下问题：

1. 克隆虚拟机并保留 BIOS UUID 和主机名时以及即时克隆 VDI 时注册失败。发生注册失败的原因是 Cisco Secure Workload 已有一个使用相同参数集的注册软件代理。您可以使用 OpenAPI 来删除已注册的代理。在某些情况下，启动时配置的重复 BIOS UUID 在一段时间后会由 VMware 更改。重启思科 Cisco Secure Workload 服务后，代理注册就会恢复。
2. 如果更改主机名并重启主机，则会为代理生成一个新身份。经过一段时间后，多余或旧的代理将被标记为不活动。有关详细信息，请参阅常见问题解答部分。

支持的平台和要求

有关支持的平台和软件代理的其他要求的信息，请参阅：

- 有关您的版本的版本说明，请参阅[版本说明](#)。
- Cisco Secure Workload Web 门户中的代理安装向导：在导航菜单中，点击**管理 (Manage)** > **工作负载 (Workloads)** > **代理 (Agents)**，然后点击**安装程序 (Installer)** 选项卡。选择安装方法、平台和代理类型（如果适用），以查看支持的平台版本。
- 其他依赖关系的[支持矩阵](#)。
- 以下各节详细介绍各平台和代理类型的其他要求。

安装用于深度可视性和执行的 Linux 代理

安装 Linux 代理的要求和前提条件

- 请参阅[支持的平台和要求](#)。
- 用于安装和执行服务的根权限。
- 1 GB 存储空间，用于代理和日志文件。
- 在监控主机的安全应用上配置安全排除项，以防止这些应用阻止代理安装或代理活动。有关详细信息，请参阅[安全排除项](#)。

- 系统会在安装代理的主机中创建一个特殊用户 **tet-sensor**。如果主机上配置了 PAM 或 SELinux，则必须授予 tet-sensor 用户执行 tet-sensor 进程和连接收集器的适当权限。如果提供了替代安装目录并配置了 SELinux，请确保允许为该位置执行。
- 如果使用 AutoInstall（安装程序脚本）方法安装代理，则必须要能够使用 `unzip` 命令。

支持的 Linux 代理安装方法

安装 Linux 代理以实现深度可视性和执行的方法：

- [使用代理脚本安装程序方法安装 Linux 代理, on page 4](#)
- [使用代理映像安装程序方法安装 Linux 代理, on page 3](#)

使用代理映像安装程序方法安装 Linux 代理

建议使用自动安装程序脚本方法来安装 Linux 代理。如果有特殊原因需要使用手动方法，请使用映像安装程序方法。

前提条件：

对于 SaaS 集群以及在具有多个租户的本地集群的非默认租户上安装代理，请在 `user.cfg` 文件中配置 `ACTIVATION_KEY` 和 `HTTPS_PROXY`。有关详细信息，请参阅 [（仅限手动安装）更新用户配置文件](#)。

要使用代理映像方法来安装 Linux 代理，请执行以下操作：

Procedure

步骤 1 导航至代理安装方法：

- 如果您是新用户，请启动“快速启动” (Quick Start) 向导，然后单击 **安装代理 (Install Agents)**。
- 在导航窗格中，选择 **管理 (Manage) > 代理 (Agents)**，然后选择 **安全程序 (Installer)** 选项卡。

步骤 2 单击 **代理映像安装程序 (Agent Image Installer)**。

步骤 3 在 **平台 (Platform)** 字段中，输入 Linux。

步骤 4 输入所需的代理类型和代理版本，然后从结果中下载所需的代理版本。

步骤 5 将 RPM 软件包复制到所有 Linux 主机进行部署。

Note 如果主机上已安装代理，请不要重新安装代理。要升级代理，请参阅升级软件代理部分。

步骤 6 使用根权限运行 RPM 命令，具体取决于您的平台。

- 对于 RHEL/CentOS/Oracle 平台，运行命令：`rpm -ivh <rpm_filename>`
- 对于 Ubuntu 平台：
 - 要检索依赖关系列表并确保满足所有依赖关系，请运行命令：`rpm -qpR <rpm_filename>`

- 通过运行以下命令，使用 “-nodeps” 选项来安装代理：`rpm -ivh \--nodeps <rpm filename>`

使用代理脚本安装程序方法安装 Linux 代理

我们建议使用安装程序脚本方法来部署 Linux 代理，以实现深度可视性和执行。



Note

- 安装的 Linux 代理支持深度可视性和执行。
- 默认情况下，执行会被禁用。要启用执行，请参阅[创建代理配置文件](#)。

要使用脚本安装程序方法来安装 Linux 代理，请执行以下操作：

Procedure

步骤 1 导航至代理安装方法：

- 如果您是新用户，请启动快速启动向导 (**Quick Start Wizard**)，然后点击**安装代理 (Install Agents)**。
- 从导航窗格中，选择**管理 (Manage) > 代理 (Agents)**，然后选择**安全程序 (Installer)** 选项卡。

步骤 2 点击代理脚本安装程序 (**Agent Script Installer**)。

步骤 3 从**选择平台 (Select Platform)** 下拉列表中选择 **Linux**。

要查看支持的 Linux 平台，请点击**显示支持的平台 (Show Supported Platforms)**。

步骤 4 选择要安装代理的租户。

Note Cisco Secure Workload SaaS 集群不需要选择租户。

步骤 5 如果要为工作负载分配标签，请选择标签键并输入标签值。

当安装的代理报告主机上的 IP 地址时，此处选择的安装程序 CMDB 标签，以及已分配给该主机报告的 IP 的其他已上传 CMDB 标签，将自动分配给新的 IP 地址。如果上传的 CMDB 标签与安装程序的 CMDB 标签发生冲突：

- 分配给确切 IP 地址的标签优先于分配给子网的标签。
- 分配给确切 IP 地址的现有标签优先于安装程序 CMDB 标签。

步骤 6 如果需要 HTTP 代理才能与 Cisco Secure Workload 通信，请选择**是 (Yes)**，然后输入有效的代理 URL。

步骤 7 在**安装程序到期 (Installer expiration)** 部分中，选择一个选项：

- 无过期 (No expiration)：安装程序脚本可多次使用。

- 一次 (One time): 安装程序脚本只能使用一次。
- 时间限制 (Time bound): 您可以设置安装程序脚本可以使用的天数。
- 部署次数 (Number of deployments): 您可以设置安装程序脚本可以使用的次数。

步骤 8 点击下载 (**Download**) 并将文件保存到本地磁盘。

步骤 9 在 Linux 主机上复制安装程序 shell 脚本并运行以下命令以授予脚本执行权限: `chmod u+x tetration_installer_default_sensor_linux.sh`

Note 脚本名称可能因所选代理类型和范围而异。

步骤 10 要安装代理, 请使用根权限运行以下命令: `./tetration_installer_default_sensor_linux.sh`

Note 如果租户上已经安装了代理, 则无法继续安装。

我们建议按照脚本使用详细信息中的规定运行预先检查。

Linux 安装程序脚本使用详细信息:

```
bash tetration_linux_installer.sh [--pre-check] [--skip-pre-check=<option>] [--no-install]
  [--logfile=<filename>] [--proxy=<proxy_string>] [--no-proxy] [--help] [--version]
  [--sensor-version=<version_info>] [--ls] [--file=<filename>] [--save=<filename>] [--new]
  [--reinstall] [--unpriv-user] [--force-upgrade] [--upgrade-local]
  [--upgrade-by-uuid=<filename>] [--basedir=<basedir>] [--logbasedir=<logbdir>]
  [--tmpdir=<tmp_dir>] [--visibility] [--golden-image]
  --pre-check: run pre-check only
  --skip-pre-check=<option>: skip pre-installation check by given option; Valid options
include 'all', 'ipv6' and 'enforcement'; e.g.: '--skip-pre-check=all' will skip all
pre-installation checks; All pre-checks will be performed by default
  --no-install: will not download and install sensor package onto the system
  --logfile=<filename>: write the log to the file specified by <filename>
  --proxy=<proxy_string>: set the value of CL_HTTPS_PROXY, the string should be formatted
as http://<proxy>:<port>
  --no-proxy: bypass system wide proxy; this flag will be ignored if --proxy flag was
provided
  --help: print this usage
  --version: print current script's version
  --sensor-version=<version_info>: select sensor's version; e.g.: '--sensor-version=3.4.1.0';
will download the latest version by default if this flag was not provided
  --ls: list all available sensor versions for your system (will not list pre-3.1 packages);
will not download any package
  --file=<filename>: provide local zip file to install sensor instead of downloading it
from cluster
  --save=<filename>: download and save zip file as <filename>
  --new: remove any previous installed sensor
  --reinstall: reinstall sensor and retain the same identity with cluster; this flag has
higher priority than --new
  --unpriv-user=<username>: use <username> for unpriv processes instead of tet-sensor
  --force-upgrade: force sensor upgrade to version given by --sensor-version flag; e.g.:
'--sensor-version=3.4.1.0 --force-upgrade'; apply the latest version by default if
--sensor-version flag was not provided
  --upgrade-local: trigger local sensor upgrade to version given by --sensor-version flag;
e.g.: '--sensor-version=3.4.1.0 --upgrade-local'; apply the latest version by default if
--sensor-version flag was not provided
  --upgrade-by-uuid=<filename>: trigger sensor whose uuid is listed in <filename> upgrade
to version given by --sensor-version flag; e.g.: '--sensor-version=3.4.1.0
--upgrade-by-uuid=/usr/local/tet/sensor_id'; apply the latest version by default if
```

```

--sensor-version flag was not provided
--basedir=<base_dir>: instead of using /usr/local use <base_dir> to install agent. The
full path will be <base_dir>/tetration
--logbasedir=<log_base_dir>: instead of logging to /usr/local/tet/log use <log_base_dir>.
The full path will be <log_base_dir>/tetration
--tmpdir=<tmp_dir>: instead of using /tmp use <tmp_dir> as temp directory
--visibility: install deep visibility agent only; --reinstall would overwrite this flag
if previous installed agent type was enforcer
--golden-image: install Cisco Secure Workload Agent but do not start the Cisco Secure
Workload Services; use to install Cisco Secure Workload Agent on Golden Images in VDI
environment or Template VM. On VDI/VM instance created from golden image with different
host name, Cisco Secure Workload Services will work normally

```

**Note**

- Ubuntu 使用本地 .deb 软件包，新安装和重装系统都会转用这种软件包类型。以前版本的升级继续使用 .rpm 软件包。
- Ubuntu .deb 软件包安装在 /opt/cisco/tetration 下。
- .deb 软件包不支持重定位，因此 Ubuntu 不支持 -basedir 选项。

验证 Linux 代理安装

Procedure

运行命令 `sudo rpm -q tet-sensor` `sudo rpm -q tet-sensor。`

```
sudo rpm -q tet-sensor
```

单个输出条目确认主机上已安装 Linux 代理。

输出示例: `tet-sensor-3.1.1.50-1.el6.x86_64`

具体输出可能会因平台和架构而异。

安装用于深度可视性和执行的 Windows 代理

安装 Windows 代理的要求和前提条件

- 请参阅“支持的平台和要求”部分。
- 安装和执行服务需要具有管理员权限。
- Npcap 必须安装在运行 Windows 2008 R2 的工作负载上，而当安装的代理版本低于版本 3.8 时必须安装。如果尚未安装 Npcap 驱动程序，代理会在服务启动后在后台安装推荐的 Npcap 版本。有关详细信息，请参阅 Npcap 版本信息。
- 1 GB 存储空间用于代理和日志文件。

- 启用代理安装所需的 Windows 服务。如果您的 Windows 主机已经过安全强化，或者偏离了默认配置，则某些 Windows 服务可能已被禁用。有关详细信息，请参阅“必需的 Windows 服务”部分。
- 在监控主机的安全应用上配置的安全排除，可能会阻止代理安装或代理活动。有关详细信息，请参阅安全排除项。

支持的 Windows 代理安装方法

有两种方法可以安装 Windows 代理，以实现深度可视性和执行。

- [使用代理脚本安装程序方法安装 Windows 代理, on page 7](#)
- [使用代理映像安装程序方法安装 Windows 代理, on page 9](#)

您还可以使用黄金映像进行安装。有关详细信息，请参阅在 [VDI 实例或 VM 模板上部署代理 \(Windows\)](#)。

使用代理脚本安装程序方法安装 Windows 代理

建议使用脚本安装程序方法来部署 Windows 代理，以实现深度可视性和执行。

**Note**

- 安装的 Windows 代理支持深度可视性和执行。
- 默认情况下，执行会被禁用。要启用执行，请参阅 [创建代理配置文件, on page 58](#)。

要使用脚本安装程序方法来安装 Windows 代理，请执行以下操作：

Procedure

步骤 1 导航至代理安装方法：

- 如果您是新用户，请启动“快速启动” (Quick Start) 向导，然后点击 **安装代理 (Install Agents)**。
- 从导航窗格中，选择 **管理 (Manage) > 代理 (Agents)**，然后选择 **安全程序 (Installer)** 选项卡。

步骤 2 点击 **代理脚本安装程序 (Agent Script Installer)**。

步骤 3 从 **选择平台 (Select Platform)** 下拉菜单中，选择 **Windows**。

要查看支持的 Windows 平台，请点击 **显示支持的平台 (Show Supported Platforms)**。

步骤 4 选择要安装代理的租户。

Note 对于 Cisco Secure Workload SaaS 集群，不需要选择租户。

步骤 5 如果要为工作负载分配标签，请选择标签键并输入标签值。

当安装的代理报告主机上的 IP 地址时，此处选择的安装程序 CMDB 标签，以及已分配给该主机报告的 IP 的其他已上传 CMDB 标签，将分配给新的 IP 地址。如果上传的 CMDB 标签与安装程序的 CMDB 标签发生冲突：

- 分配给确切 IP 地址的标签优先于分配给子网的标签。
- 分配给确切 IP 地址的现有标签优先于安装程序 CMDB 标签。

步骤 6 如果需要 HTTP 代理才能与 Cisco Secure Workload 通信，请选择是 (Yes)，然后输入有效的代理 URL。

步骤 7 在安装程序到期 (Installer expiration) 部分下，从可用选项中选择一项：

- 无过期 (No expiration)：安装程序脚本可多次使用。
- 一次 (One time)：安装程序脚本只能使用一次。
- 时间限制 (Time bound)：您可以设置安装程序脚本可以使用的天数。
- 部署次数 (Number of deployments)：您可以设置安装程序脚本可以使用的次数。

步骤 8 点击下载 (Download) 并将文件保存到本地磁盘。

步骤 9 将安装程序 PowerShell 脚本复制到所有 Windows 主机进行部署，并以管理权限运行该脚本。

- Note**
- 根据系统设置，可能需要在运行其他命令之前运行 Unblock-File 命令。
 - 如果租户上已安装代理，则脚本不会运行。

我们建议按照脚本使用详细信息中的规定运行预先检查。

Windows 安装程序脚本使用详细信息：

```
# powershell -ExecutionPolicy Bypass -File tetration_windows_installer.ps1 [-preCheck]
[-skipPreCheck <Option>] [-noInstall] [-logFile <FileName>] [-proxy <ProxyString>] [-noProxy]
[-help] [-version] [-sensorVersion <VersionInfo>] [-ls] [-file <FileName>] [-save <FileName>]
[-new] [-reinstall] [
-npcap] [-forceUpgrade] [-upgradeLocal] [-upgradeByUUID <FileName>] [-visibility]
[-goldenImage] [-installFolder <Installation Path>]
-preCheck: run pre-check only
-skipPreCheck <Option>: skip pre-installation check by given option; Valid options include
'all', 'ipv6' and 'enforcement'; e.g.: '-skipPreCheck all' will skip all pre-installation
checks; All pre-checks will be performed by default
-noInstall: will not download and install sensor package onto the system
-logFile <FileName>: write the log to the file specified by <FileName>
-proxy <ProxyString>: set the value of HTTPS_PROXY, the string should be formatted as
http://<proxy>:<port>
-noProxy: bypass system wide proxy; this flag will be ignored if -proxy flag was provided

-help: print this usage
-version: print current script's version
-sensorVersion <VersionInfo>: select sensor's version; e.g.: '-sensorVersion 3.4.1.0.win64';
will download the latest version by default if this flag was not provided
-ls: list all available sensor versions for your system (will not list pre-3.1 packages);
will not download any package
-file <FileName>: provide local zip file to install sensor instead of downloading it from
cluster
```



```

    -save <FileName>: downloaded and save zip file as <FileName>
    -new: remove any previous installed sensor; previous sensor identity has to be removed
    from cluster in order for the new registration to succeed
    -reinstall: reinstall sensor and retain the same identity with cluster; this flag has
    higher priority than -new
    -npcap: overwrite existing npcap
    -forceUpgrade: force sensor upgrade to version given by -sensorVersion flag; e.g.:
    '-sensorVersion 3.4.1.0.win64 -forceUpgrade'; apply the latest version by default if
    -sensorVersion flag was not provided
    -upgradeLocal: trigger local sensor upgrade to version given by -sensorVersion flag; e.g.:
    '-sensorVersion 3.4.1.0.win64 -upgradeLocal'; apply the latest version by default if
    -sensorVersion flag was not provided
    -upgradeByUUID <FileName>: trigger sensor whose uuid is listed in <FileName> upgrade to
    version given by -sensorVersion flag; e.g.: '-sensorVersion 3.4.1.0.win64 -upgradeByUUID
    "C:\\Program Files\\Cisco Tetration\\sensor_id"'; apply the latest version by default if
    -sensorVersion flag was not provided
    -visibility: install deep visibility agent only; -reinstall would overwrite this flag if
    previous installed agent type was enforcer
    -goldenImage: install Cisco Secure Workload Agent but do not start the Cisco Secure
    Workload Services; use to install Cisco Secure Workload Agent on Golden Images in VDI
    environment or Template VM. On VDI/VM instance created from golden image with different
    host name, Cisco Secure Workload Services will work normally
    -installFolder: install Cisco Secure Workload Agent in a custom folder specified by
    -installFolder e.g.: '-installFolder "c:\\custom sensor path"'; default path is "C:\\Program
    Files\\Cisco Tetration"

```

使用代理映像安装程序方法安装 Windows 代理

建议使用自动安装程序脚本方法来安装 Windows 代理。如果有特殊原因需要使用手动方法，请使用映像安装程序方法。



Note 当现有代理已在主机上运行时，请勿手动部署旧版 MSI 代理。

软件包中与站点相关的文件：

- **ca.cert** - 必需 - 用于传感器通信的 CA 证书。
- **enforcer.cfg** - 仅在安装执行传感器时为必需 - 包含执行终端的配置。
- **sensor_config** - 必需 - 深度可视性传感器的配置。
- **sensor_type** - 传感器的类型（执行或深度可视性）。
- **site.cfg** - 必需 - 全局站点终端配置。
- **user.cfg** - 对于 SaaS - 必需 - 传感器激活密钥和代理配置。

前提条件：

对于 SaaS 集群以及在具有多个租户的本地集群的非默认租户上安装代理，请在 **user.cfg** 文件中配置 **ACTIVATION_KEY** 和 **HTTPS_PROXY**。有关详细信息，请参阅 [（仅限手动安装）更新用户配置文件](#)。

要使用代理映像方法来安装 Windows 代理，请执行以下操作：

Procedure

步骤 1 导航至代理安装方法：

- 如果您是新用户，请启动“快速启动” (Quick Start) 向导，然后点击**安装代理 (Install Agents)**。
- 从导航窗格中，选择**管理 (Manage) > 代理 (Agents)**，然后选择**安全程序 (Installer)** 选项卡。

步骤 2 点击代理映像安装程序 (**Agent Image Installer**)。

步骤 3 在平台 (**Platform**) 字段中，输入 Windows。

步骤 4 输入所需的代理类型和代理版本，然后从结果中下载所需的代理版本。

步骤 5 将 `tet-win-sensor<version>.win64-<clustername>.zip` 文件复制到所有 Windows 主机进行部署。

步骤 6 确保您具有管理权限并提取 ZIP 文件。

步骤 7 在提取的文件夹中，运行以下命令以安装代理：`msiexec.exe /i TetratationAgentInstaller.msi`

此外，以下选项可用于 MSI 安装程序。

Table 1: MSI 安装程序的可用选项

选项	描述
<code>agenttype=<AgentType></code>	<i>AgentType</i> 应为 <i>sensor</i> 或 <i>enforcer</i> ，具体取决于是否需要执行。默认情况下，安装程序会检查同一文件夹中 <code>sensor_type</code> 文件的内容，并使用该内容来覆盖传递的参数。但是，如果是在 <i>/quiet</i> 模式下安装代理，则需要该选项。
<code>overwritenpcap=yes</code>	对于 Windows 2008 R2，默认情况下，如果 Npcap 已存在，则代理不会尝试升级 Npcap。传递此参数可升级现有 Npcap。如果使用该选项，后续的代理自动升级也会将 Npcap 升级到更新的受支持版本。
<code>nostart=yes</code>	在 VDI 环境或 VM 模板中使用黄金映像安装代理时，传递此参数可防止代理服务 TetSensor TetEnforcer 自动启动。在使用黄金镜像创建并使用不同主机名的 VDI/VM 实例上，这些服务会按预期自动启动。
<code>installfolder=<FullPathCustomFolder></code>	在安装命令末尾使用此参数，以便在自定义文件夹中安装代理。

选项	描述
serviceuser=<Service UserName>	<p>在安装命令末尾使用此参数可配置服务用户。默认服务用户为“LocalSystem”。</p> <p>对于本地用户，serviceuser=.\<Service UserName></p> <p>对于域用户，serviceuser=<domain_name>\<samaccount name></p> <p>服务用户必须拥有本地管理权限。</p> <p>服务帐户必须拥有本地管理或域管理员组权限。</p>
servicepassword=<Service UserPassword>	<p>在安装命令末尾使用此参数，以便为服务用户配置密码。密码必须为纯文本格式。</p>
proxy=" <proxy_address>"	<p>使用此参数可设置用于访问 Cisco Secure Workload 集群的 HTTPS 代理。</p>
activationkey=<activation Key>	<p>如果未在默认租户下安装代理，请使用此参数来指定租户。</p>

**Note**

- 如果在手动安装期间使用激活密钥和代理选项，则无需手动配置 *user.cfg*。
- 对于 Windows 2008 R2 以外的 Windows 操作系统，当您升级到版本 3.8 时，Windows 代理会自动卸载已安装的 Npcap。
- 如果主机上已安装代理，请不要重新安装代理。要升级代理，请参阅升级软件代理部分。

验证 Windows 代理安装

Procedure

步骤 1 确保文件夹 `C:\Program Files\Cisco Tetration`（或自定义文件夹）存在。

步骤 2 确保用于深度可视性和执行的服务 - *TetSensor* 存在且处于运行状态。使用管理权限运行命令 `cmd.exe`。

运行命令 `sc query tetsensor`

检查状态是否为正在运行 (**Running**)

运行命令 `sc qc tetsensor`

检查显示名称是否为 **Cisco Secure Workload 深度可视性 (Cisco Secure Workload Deep Visibility)**

或

运行命令 `services.msc`

查找 **Cisco Secure Workload 深度可视性 (Cisco Secure Workload Deep Visibility)** 的名称

检查状态是否为正在运行 (**Running**)

在已配置的服务用户情景中验证 Windows 代理

1. 确保服务 TetSensor（用于深度可视性）和 TetEnforcer（用于执行）在配置的服务用户情景中运行。TetSensor 和 TetEnforcer 在同一服务用户情景中运行。

使用管理员权限运行命令 `cmd.exe`

运行命令 `sc qc tetsensor`

检查 SERVICE_START_NAME <configured service user>

运行命令 `sc qc tetenforcer`

检查 SERVICE_START_NAME <configured service user>

或

运行命令 `services.msc`

查找 **Cisco Secure Workload 深度可视性 (Cisco Secure Workload Deep Visibility)** 的名称

检查 <configured service user> 的登录身份

查找名称 **Cisco Secure Workload 执行 (Cisco Secure Workload Enforcement)**

检查 <configured service user> 的登录身份

或

运行命令 `tasklist /v | find /i "tet"`

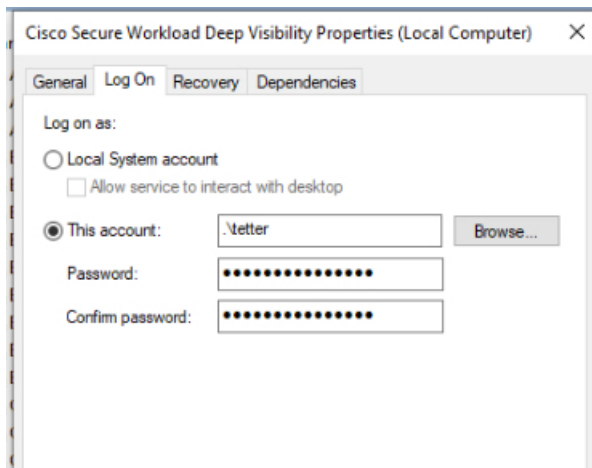
运行命令 `tasklist /v | find /i "cswengine"`

检查正在运行的进程的登录身份（第 5 列）

修改服务帐户

在安装 Windows 代理后，使用以下方法之一修改现有的深度可视性和执行服务。

- 使用 `services.msc`。

图 1: 根据 `services.msc` 帐户修改服务帐户

- 使用任何第三方应用来配置服务。
- 使用以下命令：
 1. 以管理员身份运行 `cmd`。
 2. 通过运行以下命令，使用服务帐户名称来修改服务：
 1. `sc config tetsensor obj= <service user name> password= <password>`
 2. `sc config tetenforcer obj= <service user name> password= <password>`
 3. 通过运行以下命令来验证服务配置：
 1. `sc qc tetsensor`
 2. `sc qc tetenforcer`
 4. 通过运行以下命令来重启 `tetsensor` 和 `tetenforcer` 服务：
 1. `sc stop tetsensor / tetenforcer`
 2. `sc start tetsensor / tetenforcer`

在 VDI 实例或 VM 模板上部署代理 (Windows)

默认情况下，代理服务会在代理安装后自动启动。在黄金镜像上安装时，必须使用安装程序标记来阻止这些服务启动。从黄金映像克隆实例时，代理服务会按预期自动启动。

代理不会在黄金 VM 上安装 `Npcap`，但如有需要，会在从黄金映像克隆的 VM 实例上自动安装。有关详细信息，请参阅 [Windows 代理安装程序和 Npcap - 适用于 Windows 2008 R2](#)。

在 VDI 环境或 VM 模板中的黄金映像上安装代理

Procedure

步骤 1 使用 MSI 安装程序或 PowerShell 安装程序脚本，在 VDI 环境或虚拟机模板的黄金映像上安装代理：

使用 **nostart=yes** 的 MSI 安装程序

- 有关详细信息，请参阅[使用代理映像安装程序方法安装 Windows 代理, on page 9](#)。
- `msiexec.exe /<MSI installer> nostart=" yes" /quiet /norestart /! *v <installer_log_file>` 或

或

使用带有 **-goldenImage** 标志的 PowerShell 安装程序。

- 有关详细信息，请参阅[使用代理脚本安装程序方法安装 Windows 代理, on page 7](#)。

步骤 2 确保文件夹 `C:\Program Files\Cisco Tetration`（或自定义文件夹）存在。

步骤 3 确保 TetSensor 服务（用于深度可视性）存在并且已停止：

使用**管理员**权限运行命令 `cmd.exe`。

运行命令 `sc query tetsensor`。

检查状态是否为**已停止 (Stopped)**。

步骤 4 确保服务 TetEnforcer（用于执行）存在且已停止：

运行命令 `sc query tetenforcer`。

检查状态是否为**已停止 (Stopped)**

。

步骤 5 虚拟机模板现已配置。

步骤 6 关闭 VM 模板。

创建新的 VDI 实例虚拟机

Procedure

步骤 1 通过克隆虚拟机模板创建新的 VDI 实例虚拟机。

步骤 2 重启 VDI 实例 VM。

步骤 3 重启 VDI 实例 VM 后，请确保服务 - TetSensor（用于深度可视性）和 TetEnforcer（用于执行） - 在配置的服务情景中运行。请参阅[验证 Windows 代理安装](#)。

步骤 4 在 VDI 实例虚拟机上，确保 NPCAP 驱动程序已安装并正在运行：

使用**管理员**权限运行命令 `cmd.exe`

运行命令 `sc query npcap`

检查状态是否正在运行

步骤 5 在 VDI 实例虚拟机上，确保使用有效的 `sensor_id` 来注册代理：

- 检查安装文件夹中的 `sensor_id` 文件。
- 如果 `sensor_id` 以 “`uuid`” 开头，则它不是有效的 `sensor_id`。
- 如果代理注册失败，但 Cisco Secure Workload Web 界面显示代理已注册：
- 使用 OpenAPI 删除代理。有关详细信息，请参阅 [部署软件代理](#)。

- Note**
- 请勿更改黄金映像或 VM 模板的主机名。
 - 如果黄金映像或虚拟机模板在安装代理后重启，则 Cisco Secure Workload 服务将在重启后开始运行。
 - 如果 VDI 实例虚拟机无法报告网络流，请参阅网络流中的 VDI 实例虚拟机部分。

Windows 代理安装程序和 Npcap - 适用于 Windows 2008 R2

1. 有关受支持的 Npcap 版本，请参阅 <https://www.cisco.com/go/secure-workload/requirements/agents> 上的支持矩阵。

2. 安装：

如果未安装 Npcap，代理会在服务启动十秒后安装受支持的版本。如果用户已安装 Npcap，但其版本早于支持的版本，则 Npcap 无法升级。手动升级或卸载 Npcap，使用选项 `overwrittenpcap=yes` 来运行代理安装程序，或使用 `-npcap` 来运行安装程序脚本，以获取支持的 Npcap 版本。如果任何应用正在使用 Npcap 驱动程序，则代理稍后会升级 Npcap。

3. 升级：

如果 Npcap 是由 Windows Agent 安装的，且版本早于支持的版本，则 Npcap 会在服务启动十秒后升级到支持的版本。如果任何应用正在使用 Npcap 驱动程序，则代理稍后会升级 Npcap。如果 Npcap 不是由 Windows 代理安装，则不会升级 Npcap。

4. 卸载：

如果 Npcap 是由 Windows 代理安装的，则代理会卸载 Npcap。如果 Npcap 由用户安装，但由代理安装程序使用 `overwrittenpcap=yes` 进行升级，则不会卸载 Npcap。如果任何应用正在使用 Npcap 驱动程序，则代理不会卸载 Npcap。

Windows 代理流捕获：适用于除 Windows Server 2008 R2 之外的所有 Windows 操作系统

从最新版本的 Windows 开始，该代理使用 `ndiscap.sys`（Microsoft 内置）驱动程序和使用 Windows 的事件跟踪 (ETW) 框架来捕获网络流。

在升级到最新版本期间：

- 代理从 `npcap.sys` 切换到 `ndiscap.sys`。
- 在以下情况时，代理安装程序会卸载 Npcap：
 - Npcap 由代理安装。
 - 未使用 Npcap。
 - 操作系统版本并非 Windows Server 2008 R2。

代理服务启动后，代理会创建 ETW 会话 `CSW_MonNet` 和 `CSW_MonDns`（用于 DNS 数据），同时启动网络流捕获。



Note

- 在 Windows Server 2012 上，系统会解析网络数据包以获取 DNS 数据。

安装用于深度可视性和执行的 AIX 代理



Note

进程树、软件包 (CVE) 和取证事件报告功能在 AIX 上不可用。此外，由于操作系统限制，这些功能的某些方面可能在其他支持的平台的特定次要版本上不可用。

安装 AIX 代理的要求和前提条件

- 请参阅[支持的平台和要求](#)。
- 深度可视性的其他要求：
 - 用于安装和执行服务的根权限。
 - 代理和日志文件的存储要求：500 MB。
 - 在监控主机的任何安全应用上配置的安全排除。这些排除是为了防止其他安全应用阻止代理安装或代理活动。有关详细信息，请参阅[安全排除项](#)。
 - AIX 仅支持 20 台网络设备的流捕获（如果版本为 AIX 7.1 TL3 SP4 或更早，则为 6 台网络设备）。深度可视性代理最多从 16 台网络设备捕获，而其他 4 个捕获会话可用于专用的通用系统（例如，`tcpdump`）。
 - 深度可视性代理执行以下操作，以确保对 20 台网络设备进行流捕获：
 - 代理会在代理目录 (`/opt/cisco/tetration/chroot/dev/bpf0 - /opt/cisco/tetration/chroot/dev/bpf15`) 下创建 16 个 bpf 设备节点
 - `tcpdump` 和其他使用 bpf 的系统工具将扫描系统设备节点 (`/dev/bpf0- /dev/bpf19`)，直至找到未使用的节点 (`!EBUSY`)

- 代理创建的 bpf 节点和系统 bpf 节点会共享相同的主/次要文件，而每个主要或次要文件仅由一个实例（tcpdump 或代理）打开。
- 代理不会访问系统设备节点，也不会像 tcpdump 那样创建系统设备节点（如果不存在，tcpdump-D 会创建 /dev/bpf0.../dev/bpf19）。
- 在某些情况下，在系统上运行 iptrace 可防止来自 tcpdump 和深度可视性代理的流捕获。这是一个已知的设计问题，需要与 IBM 核实。
 - 要检查是否存在此场景，请在安装代理之前运行 tcpdump。如果错误消息为 **tcpdump: BIOCSSETIF: en0: File exists**，则 iptrace 已阻止流捕获。停止 iptrace 以解决问题。
- 并非所有深度可视性功能在 AIX 中都不受支持。数据包和进程记帐属于不支持的类型。
- 策略执行的其他要求：
 - 如果启用了 IP 安全过滤器（即 smitty IPsec4），则代理安装会在预先检查中失败。建议您在安装代理之前禁用 IP 安全过滤器。
 - 如果在运行 Cisco Secure Workload 执行器代理时启用了 IP 安全，则会报告错误并停止执行。请联系支持人员，以便在执行器代理运行时安全地禁用 IP 安全过滤器。

使用代理脚本安装程序方法安装 AIX 代理

深度可视性和执行 AIX 代理只能使用代理脚本安装方法进行安装。



Note

- 安装的 AIX 代理支持深度可视性和执行。
- 默认情况下，执行会被禁用。要启用执行，请参阅[创建代理配置文件, on page 58](#)。

要安装 AIX 代理，请执行以下操作：

Procedure

- 步骤 1** 导航至代理安装方法：
 - 如果您是新用户，请启动“快速启动”（Quick Start）向导，然后单击**安装代理 (Install Agents)**。
 - 从导航窗格中，选择**管理 (Manage) > 代理 (Agents)**，然后选择**安全程序 (Installer)** 选项卡。
- 步骤 2** 单击代理脚本安装程序 (**Agent Script Installer**)。
- 步骤 3** 从**选择平台 (Select Platform)** 下拉菜单中，选择 **AIX**。
要查看支持的 AIX 平台，请点击**显示支持的平台 (Show Supported Platforms)**。
- 步骤 4** 选择要安装代理的租户。

Note 对于 Cisco Secure Workload SaaS 集群，不需要选择租户。

步骤 5 如果要为工作负载分配标签，请选择标签键并输入标签值。

当安装的代理报告主机上的 IP 地址时，此处选择的安装程序 CMDB 标签，以及已分配给该主机报告的 IP 的其他已上传 CMDB 标签，将自动分配给新的 IP 地址。如果上传的 CMDB 标签与安装程序的 CMDB 标签发生冲突：

- 分配给确切 IP 地址的标签优先于分配给子网的标签。
- 分配给确切 IP 地址的现有标签优先于安装程序 CMDB 标签。

步骤 6 如果需要 HTTP 代理才能与 Cisco Secure Workload 通信，请选择是 (Yes)，然后输入有效的代理 URL。

步骤 7 在安装程序到期 (Installer expiration) 部分下，从可用选项选择一个选项：

- 无过期 (No expiration)：安装程序脚本可多次使用。
- 一次 (One time)：安装程序脚本只能使用一次。
- 时间限制 (Time bound)：您可以设置安装程序脚本可以使用的天数。
- 部署次数 (Number of deployments)：您可以设置安装程序脚本可以使用的次数。

步骤 8 点击下载 (Download) 并将文件保存到本地磁盘。

步骤 9 将安装程序 shell 脚本复制到所有 AIX 主机上进行部署。

步骤 10 要授予脚本执行权限，请运行以下命令：`chmod u+x tetration_installer_default_sensor_aix.sh`

Note 脚本名称可能因代理类型和范围而异。

步骤 11 要安装代理，请使用根权限运行以下命令：`./tetration_installer_default_sensor_aix.sh`

Note 如果主机上已经安装了代理，则无法继续安装。

我们建议按照脚本使用详细信息中的规定运行预先检查。

AIX 安装程序脚本使用详细信息：

```
ksh tetration_installer_default_enforcer_aix.sh [--pre-check] [--pre-check-user]
[--skip-pre-check=<option>] [--no-install] [--logfile=<filename>] [--proxy=<proxy_string>]
[--no-proxy] [--help] [--version] [--sensor-version=<version_info>] [--ls]
[--file=<filename>] [--osversion=<osversion>] [--save=<filename>] [--new] [--reinstall]
[--unpriv-user] [--libs=<libs.zip|tar.Z>] [--force-upgrade] [--upgrade-local]
[--upgrade-by-uuid=<filename>] [--logbasedir=<logbdir>] [--tmpdir=<tmp_dir>] [--visibility]
[--golden-image]
--pre-check: run pre-check only
--pre-check-user: provide alternative to nobody user for pre-check su support
--skip-pre-check=<option>: skip pre-installation check by given option; Valid options
include 'all', 'ipv6' and 'enforcement'; e.g.: '--skip-pre-check=all' will skip all
pre-installation checks; All pre-checks will be performed by default
--no-install: will not download and install sensor package onto the system
--logfile=<filename>: write the log to the file specified by <filename>
--proxy=<proxy_string>: set the value of HTTPS_PROXY, the string should be formatted as
```

```

http://<proxy>:<port>
--no-proxy: bypass system wide proxy; this flag will be ignored if --proxy flag was
provided
--help: print this usage
--version: print current script's version
--sensor-version=<version_info>: select sensor's version; e.g.: '--sensor-version=3.4.1.0';
will download the latest version by default if this flag was not provided
--ls: list all available sensor versions for your system (will not list pre-3.3 packages);
will not download any package
--file=<filename>: provide local zip file to install sensor instead of downloading it
from cluster
--osversion=<osversion>: specify osversion for --save flag;
--save=<filename>: download and save zip file as <filename>; will download package for
osversion given by --osversion flag; e.g.: '--save=myimage.aix72.tar.Z --osversion=7.2'
--new: remove any previous installed sensor; previous sensor identity has to be removed
from cluster in order for the new registration to succeed
--reinstall: reinstall sensor and retain the same identity with cluster; this flag has
higher priority than --new
--unpriv-user=<username>: use <username> for unpriv processes instead of tet-snsr
--libs=<libs.zip|tar.Z>: install provided libs to be used by agents
--force-upgrade: force sensor upgrade to version given by --sensor-version flag; e.g.:
'--sensor-version=3.4.1.0 --force-upgrade'; apply the latest version by default if
--sensor-version flag was not provided
--upgrade-local: trigger local sensor upgrade to version given by --sensor-version flag;
e.g.: '--sensor-version=3.4.1.0 --upgrade-local'; apply the latest version by default if
--sensor-version flag was not provided
--upgrade-by-uuid=<filename>: trigger sensor whose uuid is listed in <filename> upgrade
to version given by --sensor-version flag; e.g.: '--sensor-version=3.4.1.0
--upgrade-by-uuid=/usr/local/tet/sensor_id'; apply the latest version by default if
--sensor-version flag was not provided
--logbasedir=<log_base_dir>: instead of logging to /opt/cisco/tetration/log use
<log_base_dir>. The full path will be <log_base_dir>/tetration
--tmpdir=<tmp_dir>: instead of using /tmp use <tmp_dir> as temp directory
--visibility: install deep visibility agent only; --reinstall would overwrite this flag
if previous installed agent type was enforcer
--golden-image: install Cisco Secure Workload Agent but do not start the Cisco Secure
Workload Services; use to install Cisco Secure Workload Agent on Golden Images in VDI
environment or Template VM. On VDI/VM instance created from golden image with different
host name, Cisco Secure Workload Services will work normally

```

验证 AIX 代理安装

过程

运行命令 `lsllpp -c -l tet-sensor.rte`，确认存在如下的条目。

注释 具体输出可能因版本而异

```
$ sudo lsllpp -c -l tet-sensor.rte /usr/lib/objrepos:tet-sensor.rte:3.4.1.19::COMMITTED:I:TET tet
sensor package:
```

```
$ sudo lssrc -s tet-sensor
```

```
Subsystem Group PID Status tet-sensor 1234567 active
```

```
$ sudo lssrc -s tet-enforcer
```

Subsystem Group PID Status tet-enforcer 7654321 active

安装 Kubernetes 或 OpenShift 代理以实现深度可视性和执行

要求和前提条件

操作系统支持信息可在[代理 OS 支持矩阵 \(Agent OS support matrix\)](#)中找到。

要求

- 安装脚本需要 Kubernetes 或 OpenShift 管理员凭证，以便在集群节点上启动特权代理 Pod。
- Cisco Secure Workload 实体在 **tetration** 命名空间中创建。
- 节点或 Pod 安全策略必须允许特权模式 Pod。
- busybox:1.33 映像必须预安装或可从 Docker Hub 下载。
- 对于容器化的行时，如果未设置 `config_path`，请修改 `config.toml`（默认位置：`/etc/containerd/config.toml`），如下所示：

```

...
    [plugins."io.containerd.grpc.v1.cri".registry]
      config_path = "/etc/containerd/certs.d"
...

```

重启 `containerd` 后台守护程序。

- 要在 Kubernetes 或 OpenShift 控制平面节点上运行，`-toleration` 标志可用于传入 Cisco Secure Workload Pod 的容差。通常通过的容差是 `NoSchedule` 容差，通常会阻止 Pod 在控制平面节点上运行。
- 对于 Windows 工作节点：
 - 支持的 Windows 工作节点容器运行时：ContainerD。
 - ContainerD 配置：配置以下 `containerd` 更改。

```

...
    [plugins."io.containerd.grpc.v1.cri".registry]
      config_path = "/etc/containerd/certs.d"
...

```

删除 **registry.mirrors** 下的配置。默认配置文件位置为 `C:\Program Files\containerd\config.toml`。

在配置更改后，重启 `containerd` 后台守护程序。

- 映像 mcr.microsoft.com/oss/kubernetes/windows-host-process-containers-base-image:v1.0.0 必须在 Windows 工作节点上预安装或可下载。

- 正在升级到较新版本的现有 Kubernetes 代理会自动包含 Windows 守护进程集代理。但是，前一个脚本不会卸载 Windows 守护进程集代理。下载最新的安装程序脚本，以便卸载 Windows 守护进程集代理。
- 支持的型号：
 - Microsoft Windows Server 2022
 - Windows Server 2019
 - Kubernetes 1.27 及更高版本

策略执行要求

OpenShift 不支持基于 IPVS 的 kube-proxy 模式。

这些代理应配置为启用保留规则选项。有关更多信息，请参阅[创建代理配置文件](#)。

为使执行正常运行，任何已安装的 CNI 插件都必须：

- 在所有节点和 Pod 之间提供平面地址空间（IP 网络）。不支持伪装源 Pod IP 以进行集群内通信的网络插件。
- 不干扰 Cisco Secure Workload 执行代理使用的 Linux iptables 规则或标记（标记位 21 和 20 用于允许和拒绝 NodePort 服务的流量）

以下 CNI 插件经过测试，可满足上述要求：

- Calico (3.13) 与以下 Felix 配置：*(ChainInsertMode: Append, IptablesRefreshInterval: 0)* 或 *(ChainInsertMode: Insert, IptablesFilterAllowAction: Return, IptablesMangleAllowAction: Return, IptablesRefreshInterval: 0)*。所有其他选项均使用各自的默认值。

有关设置这些选项的详细信息，请参阅 [Felix 配置参考](#)。

使用代理脚本安装程序方法安装 Kubernetes 或 OpenShift 代理



Note 代理脚本安装程序方法会在稍后包含的节点上自动安装代理。

Procedure

步骤 1 导航至代理安装方法：

- 如果您是新用户，请启动“快速启动” (Quick Start) 向导，然后单击**安装代理 (Install Agents)**。
- 从导航窗格中，选择**管理 (Manage) > 代理 (Agents)**，然后选择安全程序 (**Installer**) 选项卡。

步骤 2 单击代理脚本安装程序 (**Agent Script Installer**)。

步骤 3 从选择平台 (**Select Platform**) 下拉菜单中, 选择 **Kubernetes**。

要查看支持的 Kubernetes 或 OpenShift 平台, 请点击显示支持的平台 (**Show Supported Platforms**)。

步骤 4 选择要安装代理的租户。

Note 对于 Cisco Secure Workload SaaS 集群, 不需要选择租户。

步骤 5 如果需要 HTTP 代理才能与 Cisco Secure Workload 通信, 请选择是 (**Yes**), 然后输入有效的代理 URL。

步骤 6 点击下载 (**Download**) 并将文件保存到本地磁盘。

步骤 7 在 Linux 机器上运行安装程序脚本, 该机器可访问 Kubernetes API 服务器和具有管理权限的 kubectl 配置文件, 并将其作为默认上下文/集群/用户。

安装程序会尝试从默认位置 (~/.kube/config) 读取文件。但是, 您可以使用 --kubeconfig 命令来明确指定配置文件的位置。

安装脚本提供了验证 Cisco Secure Workload 代理守护进程集和已安装 Pod 的说明。



Note 下载前在代理安装程序页面上配置的 HTTP 代理仅控制 Cisco Secure Workload 代理连接到 Cisco Secure Workload 集群的方式。此设置不会影响 Kubernetes 或 OpenShift 节点获取 Docker 映像的方式, 因为这些节点上的容器运行时会使用自己的代理配置。如果没有从 Cisco Secure Workload 集群获取 Docker 映像, 请调试容器的映像提取过程, 并添加合适的 HTTP 代理。

安装用于深度可视性和执行的 Solaris 代理

安装 Solaris 代理的要求和前提条件

- 请参阅[支持的平台和要求](#)。
- 用于安装和执行服务的根权限。
- 1 GB 存储空间用于代理和日志文件。
- 在监控主机的安全应用上配置安全排除, 以防止其他安全应用阻止代理安装或代理活动。有关详细信息, 请参阅[安全排除项](#)。

使用代理脚本安装程序方法安装 Solaris 代理

已安装的 Solaris 代理支持深度可视性和进程或软件包可视性。

Procedure

步骤 1 导航至代理安装方法：

- 如果您是新用户，请启动“快速启动”(Quick Start)向导，然后点击**安装代理 (Install Agents)**。
- 从导航窗格中，选择**管理 (Manage) > 代理 (Agents)**，然后选择**安全程序 (Installer)**选项卡。

步骤 2 点击**代理脚本安装程序 (Agent Script Installer)**。

步骤 3 从**选择平台 (Select Platform)**下拉菜单中，选择**Solaris**。

要查看支持的 Solaris 平台，请点击**显示支持的平台 (Show Supported Platforms)**。

步骤 4 选择要安装代理的租户。

Note Cisco Secure Workload SaaS 集群不需要选择租户。

步骤 5 如果要为工作负载分配标签，请选择标签键并输入标签值。

当已安装的代理报告主机上的 IP 地址时，此处选择的安装程序 CMDB 标签以及已分配给此主机报告的 IP 的其他上传 CMDB 标签将自动分配给新的 IP 地址。当安装的代理报告主机上的 IP 地址时，此处选择的安装程序 CMDB 标签，以及已分配给该主机报告的 IP 的其他已上传 CMDB 标签，将被自动分配给新的 IP 地址。如果上传的 CMDB 标签与安装程序的 CMDB 标签发生冲突：

- 分配给确切 IP 地址的标签优先于分配给子网的标签。
- 分配给确切 IP 地址的现有标签优先于安装程序 CMDB 标签。

步骤 6 如果需要 HTTP 代理才能与 Cisco Secure Workload 通信，请选择**是 (Yes)**，然后输入有效的代理 URL。

步骤 7 在**安装程序到期 (Installer expiration)**部分下，从可用选项中选择一项：

- **无过期 (No expiration)**：安装程序脚本可多次使用。
- **一次 (One time)**：安装程序脚本只能使用一次。
- **时间限制 (Time bound)**：您可以设置安装程序脚本可以使用的天数。
- **部署次数 (Number of deployments)**：您可以设置安装程序脚本可以使用的次数。

步骤 8 点击**下载 (Download)**并将文件保存到本地磁盘。

步骤 9 在 Solaris 主机上复制安装程序 shell 脚本并运行以下命令以授予脚本执行权限：`chmod u+x tetration_installer_default_sensor_solaris.sh`

Note 脚本名称可能因所选代理类型和范围而异。

步骤 10 要安装代理，请使用根权限运行以下命令：`./tetration_installer_default_sensor_solaris.sh`

Note 如果租户上已经安装了代理，则无法继续安装。

我们建议按照脚本使用详细信息中的规定运行预先检查。

Solaris 安装程序脚本使用详细信息：

```
tetration_installer_default_sensor_solaris.sh [--pre-check] [--skip-pre-check=<option>]
[--no-install] [--logfile=<filename>] [--proxy=<proxy_string>] [--no-proxy] [--help]
[--version] [--sensor-version=<version_info>] [--ls] [--file=<filename>] [--save=<filename>]
[--new] [--reinstall] [--unpriv-user] [--force-upgrade] [--upgrade-local]
[--upgrade-by-uuid=<filename>] [--basedir=<basedir>] [--logbasedir=<logbdir>]
[--tmpdir=<tmp_dir>] [--visibility] [--golden-image]
--pre-check: run pre-check only
--skip-pre-check=<option>: skip pre-installation check by given option; Valid options
include 'all', 'ipv6' and 'enforcement'; e.g.: '--skip-pre-check=all' will skip all
pre-installation checks; All pre-checks will be performed by default
--no-install: will not download and install sensor package onto the system
--logfile=<filename>: write the log to the file specified by <filename>
--proxy=<proxy_string>: set the value of CL_HTTPS_PROXY, the string should be formatted
as http://<proxy>:<port>
--no-proxy: bypass system wide proxy; this flag will be ignored if --proxy flag was
provided
--help: print this usage
--version: print current script's version
--sensor-version=<version_info>: select sensor's version; e.g.: '--sensor-version=3.4.1.0';
will download the latest version by default if this flag was not provided
--ls: list all available sensor versions for your system (will not list pre-3.1 packages);
will not download any package
--file=<filename>: provide local zip file to install sensor instead of downloading it
from cluster
--save=<filename>: download and save zip file as <filename>
--new: remove any previous installed sensor; previous sensor identity has to be removed
from cluster in order for the new registration to succeed
--reinstall: reinstall sensor and retain the same identity with cluster; this flag has
higher priority than --new
--unpriv-user=<username>: use <username> for unpriv processes instead of nobody
--force-upgrade: force sensor upgrade to version given by --sensor-version flag; e.g.:
'--sensor-version=3.4.1.0 --force-upgrade'; apply the latest version by default if
--sensor-version flag was not provided
--upgrade-local: trigger local sensor upgrade to version given by --sensor-version flag;
e.g.: '--sensor-version=3.4.1.0 --upgrade-local'; apply the latest version by default if
--sensor-version flag was not provided
--upgrade-by-uuid=<filename>: trigger sensor whose uuid is listed in <filename> upgrade
to version given by --sensor-version flag; e.g.: '--sensor-version=3.4.1.0
--upgrade-by-uuid=/usr/local/tet/sensor_id'; apply the latest version by default if
--sensor-version flag was not provided
--logbasedir=<log_base_dir>: instead of logging to /opt/cisco/secure-workload/log use
<log_base_dir>. The full path will be <log_base_dir>/secure-workload
--tmpdir=<tmp_dir>: instead of using /tmp use <tmp_dir> as temp directory
--visibility: install deep visibility agent only; --reinstall would overwrite this flag
if previous installed agent type was enforcer
--golden-image: install Cisco Secure Workload Agent but do not start the Cisco Secure
Workload Services; use to install Cisco Secure Workload Agent on Golden Images in VDI
environment or Template VM. On VDI/VM instance created from golden image with different
host name, Cisco Secure Workload Services will work normally
```

验证 Solaris 代理安装

- 对于 Solaris 11.4，请运行命令 `sudo pkg list tet-sensor`

单个输出条目确认主机上已安装 Solaris 代理。以下是示例输出：

名称 (发布服务器)	版本	IFO
tet-sensor (cisco)	3.8.1.1	i--

(仅限手动安装) 更新用户配置文件

只有涉及以下所有内容的安装才需要执行以下程序：

- Cisco Secure Workload SaaS 或具有多个租户的本地集群（仅使用默认租户的本地集群不需要此程序）
- 手动安装
- Linux 或 Windows 平台

代理需要激活密钥才能注册到 Cisco Secure Workload 集群。它们需要使用集群激活密钥。此外，他们可能需要使用 HTTPS 代理来访问集群。



Note 在 Windows 环境中，如果手动安装时使用了激活密钥和代理选项，则无需手动配置 `user.cfg`。

在安装前，请在用户配置文件中配置所需的变量：

Procedure

- 步骤 1** 要检索激活密钥，请导航至管理 (Manage) > 代理 (Agents)，点击安装程序 (Installer) 选项卡，点击使用经典打包安装程序手动安装 (Manual Install using classic packaged installers)，然后点击代理激活密钥 (Agent Activation Key)。
- 步骤 2** 打开 Cisco Secure Workload 代理安装文件夹中的 `user.cfg` 文件。（例如：Linux 上的 `/usr/local/tet` 或 Windows 上的 `C:\Program Files\Cisco Tetration`）。该文件包含 “key=value” 形式的变量列表，每行一个。
- 步骤 3** 将激活密钥添加到 `ACTIVATION_KEY` 变量。示例：
`ACTIVATION_KEY=7752163c635ef62e6568e9e852d07bd21bfd60d0`
- 步骤 4** 如果代理需要 HTTPS 代理，请使用 `HTTPS_PROXY` 变量来添加 `http` 协议代理服务器和端口。示例：
`HTTPS_PROXY=http://proxy.my-company.com:80`

其他类代理工具

AnyConnect 代理

具有网络可视性模块 (NVM) 的思科 AnyConnect 安全移动代理支持的平台不需要 Secure Workload 代理。AnyConnect 连接器会注册这些代理，并将流观察结果、资产和标签导出到 Cisco Secure Workload。有关详细信息，请参阅 [AnyConnect 连接器](#)。

对于 Windows、Mac 或 Linux 平台，请参阅《[思科 AnyConnect 安全移动客户端产品手册](#)》。

ISE 代理

向 Cisco Identity Services Engine (ISE) 注册的终端不需要在终端上安装 Cisco Secure Workload 代理。ISE 连接器通过 ISE 设备上的 pxGrid 服务从 ISE 收集有关终端的元数据。它将终端注册为 Cisco Secure Workload 上的 ISE 代理，并推送这些终端上资产的标签。有关详细信息，请参阅 [ISE 连接器](#)。

SPAN 代理

SPAN 代理与 ERSPAN 连接器配合使用。有关详细信息，请参阅 [ERSPAN 连接器](#)。

第三方和其他思科产品

- 对于使用 Cisco Secure Workload 中配置的外部协调器的集成，请参阅 [Cisco Secure Workload 中的外部协调器](#)。
- 对于使用在 Cisco Secure Workload 中配置的连接器的集成，请参阅 [什么是连接器](#)。

连接信息

在工作负载上安装代理时，它通常会与 Cisco Secure Workload 集群上托管的后端服务建立多个网络连接。连接数会因代理类型及其功能而异。

下表记录了各种代理类型建立的各种永久连接。

Table 2: 代理连接

代理类型	配置服务器	收集器	执行后端
可视性 (本地)	CFG-SERVER-IP:443	COLLECTOR-IP:5640	不适用
可视性 (SaaS)	CFG-SERVER-IP:443	COLLECTOR-IP:443	不适用
安全策略 (本地)	CFG-SERVER-IP:443	COLLECTOR-IP:5640	ENFORCER-IP:5660

代理类型	配置服务器	收集器	执行后端
执行 (SaaS)	CFG-SERVER-IP:443	COLLECTOR-IP:443	ENFORCER-IP:443
docker 映像	CFG-SERVER-IP:443	不适用	不适用

图例：

- CPG-SERVER-IP 是配置服务器的 IP 地址。
- COLLECTOR-IP 是收集器的 IP 地址。深度可视性和执行代理连接到所有可用的收集器。
- ENFORCER-IP 是执行终端的 IP 地址。执行代理仅连接到其中一个可用终端。
- 对于 Kubernetes/OpenShift 代理部署，安装脚本不包含代理软件 - 包含代理软件的 Docker 映像由每个 Kubernetes/OpenShift 节点从 Cisco Secure Workload 集群提取。这些连接由容器运行时映像获取组件建立，并指向 CFG-SERVER-IP:443。

导航至平台 (**Platform**) > 集群配置 (**Cluster Configuration**)，了解配置服务器 IP 和收集器 IP。

- 传感器 VIP (**Sensor VIP**) 用于配置服务器 IP：已为此集群中的配置服务器设置的 IP 地址。
- 外部 IP 用于收集器 IP 和执行器 IP：如果填充此字段，则在分配外部集群 IP 地址时，选择过程仅限于此列表中定义的属于外部网络的 IP 地址。



Note

- Cisco Secure Workload 代理始终充当客户端以发起与集群内托管的服务的连接，并且从不会作为服务器打开连接。
- 支持升级的代理会定期向集群传感器 VIP 执行 HTTPS 请求（端口 443），以查询可用的软件包。
- 代理可以位于 NAT 服务器后面。

如果工作负载位于防火墙后面，或者主机防火墙服务已启用，则到集群的连接可能会被拒绝。在此情况下，管理员必须创建适当的防火墙策略来允许连接。

安全排除项

软件代理在正常运行过程中会不断与主机操作系统交互。此操作可能会导致主机上安装的其他安全应用（例如防病毒软件、安全代理等）发出警报，或者阻止 Cisco Secure Workload 代理的操作。因此，为确保代理安装成功并正常运行，必须在监控主机的安全应用上配置必要的安全排除。

Table 3: 代理目录的安全排除项

主机操作系统	目录
AIX	/opt/cisco/tetration

主机操作系统	目录
Linux	/usr/local/tet 或 /opt/cisco/tetration 或 <user chosen inst dir>
	/var/opt/cisco/secure-workload
Windows 的 ISE 安全评估代理	C:\Program Files\Cisco Tetration
	C:\ProgramData\Cisco Tetration
Solaris	/opt/cisco/secure-workload

Table 4: 代理进程的安全排除项

主机操作系统	进程
AIX	tet-engine、tet-sensor、tet-enforcer
Linux	tet-engine、tet-sensor、tet-enforcer、tet-main、enforcer
Windows 的 ISE 安全评估代理	TetSenEngine.exe、TetSen.exe、TetEnfEngine.exe、TetEnfC.exe、TetEnf.exe、TetUpdate.exe、tet-main.exe

Table 5: 代理进程的安全排除项

主机操作系统	进程
AIX	csw-agent
	tet-sensor
	tet-enforcer
	tet-main
Linux	csw-agent
	tet-sensor
	tet-enforcer
	tet-main
	执行器

主机操作系统	进程
Windows 的 ISE 安全评估代理	TetSenEngine.exe
	TetSen.exe
	TetEnfEgine.exe
	TetEnfC.exe
	TetEnf.exe
	TetUpdate.exe
	tet-main.exe
Solaris	csw-agent
	tet-sensor
	tet-main

Table 6: 代理操作的安全排除项

主机操作系统	操作
AIX	访问 /dev/bpf*、/dev/ip1、/dev/kmem
	调用 cfg_ipf、curl、ipf、ippool、ipfstat lslpp、lsfilt、prtconf
	扫描 /proc
Linux	调用 curl、ip[6]tables-save、ip[6]tables-restore、rpm/dpkg
	扫描 /proc，打开 netlink 套接字
Windows 的 ISE 安全评估代理	访问注册表
	注册到防火墙事件
	调用 c:\windows\system32\netsh.exe
Solaris 11.4	调用 curl、lspp、pkg、smbios
	扫描 /proc
扫描 /proc	

Table 7: 代理脚本或二进制文件执行的安全排除项

主机操作系统	调用的脚本/二进制文件
AIX	-
Linux	-
Windows 的 ISE 安全评估代理	dmidecode.exe
	npcap-installer.exe
	sensortools.exe
	signtool.exe
Solaris	-

代理的服务管理

软件代理作为一项服务部署在所有支持的平台上。本部分介绍管理各种功能和平台的服务的方法。



Note 除非另有说明，否则此部分中的所有命令都需要 Linux 或 Unix 上的根权限或 Windows 上的管理权限方可运行。

RHEL、CentOS、OracleLinux-6.x 和 Ubuntu-14 的服务管理

为以下各项运行命令：

- 启动服务：`start csw-agent`
- 停止服务：`stop csw-agent`
- 重启服务：`restart csw-agent`
- 检查服务状态：`status csw-agent`

适用于 RHEL、CentOS、OracleLinux-7.x 及更高版本的服务管理

命令也适用于：

- AlmaLinux, Rocky Linux-8.x 及更高版本
- Amazon Linux 2 及更高版本
- Debian 8 及更高版本

- SLE-12SPx 及更高版本
- Ubuntu-16.04 及更高版本

为以下各项运行命令：

- 启动服务：`systemctl start csw-agent`
- 停止服务：`systemctl stop csw-agent`
- 重启服务：`systemctl restart csw-agent`
- 检查服务状态：`systemctl status csw-agent`

Windows Server 或 Windows VDI 服务管理

为以下各项运行命令：

- 启动服务：`net start <service-name>`
示例：用于深度可视性服务的 `net start tetsensor` - 用于执行服务的 `net start tetenforcer`
- 停止服务：`net stop <service-name>`
示例：用于深度可视性服务的 `net stop tetsensor` - 用于执行服务的 `net stop tetenforcer`
- 重启服务：
 1. `net stop <service-name>`
 2. `net start <service-name>`
- 检查服务状态：`sc query <service-name>`
示例：用于深度可视性服务的 `sc query tetsensor` - 用于执行服务的 `sc query tetenforcer`

AIX 的服务管理

为以下各项运行命令：

- 启动服务：`startsrc -s csw-agent`
- 停止服务：`stopsrc -s csw-agent`
- 重启服务：
 1. `stopsrc -s csw-agent`
 2. `startsrc -s csw-agent`
- 正在检查服务状态：`lssrc -s csw-agent`

Kubernetes 代理安装的服务管理

- **启动或停止服务：**无法在特定节点上启动或停止代理，因为它们不是作为单个服务安装的，而是作为整个集群的后台守护程序集安装的。
- **在节点上重启代理：**找到节点上的 Cisco Secure Workload 代理 Pod，并运行相应的 Kubernetes 命令将其终止。Pod 将自动重启。
- **检查 Pod 的状态：**`kubect1 get pod -n tetration` 或 `oc get pod -n tetration`（适用于 OpenShift）会列出 Kubernetes 集群中所有 Cisco Secure Workload 代理 Pod 的状态。

Solaris 服务管理

为以下各项运行命令：

- **启动服务：**`svcadm enable csw-agent`
- **停止服务：**`svcadm disable csw-agent`
- **重启服务：**`svcadm restart csw-agent`
- **检查服务状态：**`svcs -l csw-agent`

使用代理的执行策略

默认情况下，安装在工作负载上的代理具有执行策略的功能，但执行已被禁用。准备就绪后，您就可以启用这些代理，根据配置的意图在选定的主机上执行策略。

当代理执行策略时，它会应用一组有序的规则，根据源、目标、端口、协议和方向等参数，指定防火墙应 ALLOW 还是 DROP 特定网络流量。有关策略的详细信息，请参阅在 [Cisco Secure Workload 中管理策略生命周期](#)。

使用代理来执行

- 代理通过安全的 TCP 或 SSL 通道来接收策略。
- 代理在特权域中运行。在 Linux 计算机上，代理以 root 身份运行；在 Windows 计算机上，代理以 SYSTEM 身份运行。
- 根据平台的不同，启用策略执行后，代理可以完全控制防火墙，也可以使用现有的配置规则。
- 有关执行选项以及启用和配置代理以执行策略的详细信息，请参阅 [创建代理配置文件, on page 58](#)。

高级详细信息

启用执行后，将制定黄金规则，允许代理连接到控制器。代理通过使用 TLS 或 SSL 协议的双向安全通道与控制器的执行前端 (EFE) 进行通信。来自控制器的消息由策略生成器签名，并由代理验证。

代理从控制器接收平台无关模式的策略。代理会将这些独立于平台的策略转换为特定于平台的策略，并在终端上对防火墙进行编程。

代理会主动监控防火墙状态。如果代理检测到执行的策略有任何偏差，它会再次将缓存的策略执行到防火墙中。代理还会监控自身对 CPU 和内存等系统资源的消耗情况。

代理使用 EFE 定期向控制器发送状态和统计信息报告。状态报告包括最新编程策略的状态，如成功、失败或错误（如有）。统计信息报告包括策略统计，如允许和丢弃的数据包，以及字节数，具体取决于平台。

Linux 平台上的代理执行

在 Linux 平台上，代理使用 iptables、ip6tables 或 ipset 来执行网络策略。在主机上启用代理后，默认情况下，它会控制和编程 iptables。如果启用了 IPv6 网络堆栈，则代理会使用 ip6tables 来控制 IPv6 防火墙。

Linux iptables 或 ip6tables

Linux 内核具有 iptables 和 ip6tables，用于设置、维护和检查 IPv4 和 IPv6 数据包过滤规则表。iptables 和 ip6tables 包含许多预定义的表。每个表都包含预定义的链，也可以包含用户定义的链。这些链包含规则集，而每个规则指定了数据包的匹配条件。预定义表包括 raw、mhole、filter 和 NAT。预定义链包括 INPUT、OUTPUT、FORWARD、PREROUTING 和 POSTROUTING。

Cisco Secure Workload 代理对包含允许或丢弃数据包的规则的过滤表进行编程。过滤器表由预定义链 INPUT、OUTPUT 和 FORWARD 组成。除此以外，代理还会添加自定义 TA 链，以便对来自控制器的策略进行分类和管理。这些 TA 链包含从策略派生的 Cisco Secure Workload 规则以及代理生成的规则。当代理收到与平台无关的规则时，它会将其解析并转换为 iptable、ip6table 或 ipset 规则，并将这些规则插入 TA 在过滤表中定义的链中。在对防火墙进行编程后，代理会监控防火墙是否有任何规则或策略偏差，如有，则会重新对防火墙进行编程。它会跟踪防火墙中编程的策略，并定期向控制器报告其统计信息。

以下是一个描述这种行为的示例：

独立于平台的网络策略消息中的典型策略包括：

```
source set id: "test-set-1"
destination set id: "test-set-2"
source ports: 20-30
destination ports: 40-50
ip protocol: TCP
action: ALLOW
. . .
set_id: "test-set-1"
  ip_addr: 1.2.0.0
  prefix_length: 16
  address_family: IPv4
set_id: "test-set-2"
  ip_addr: 3.4.0.0
  prefix_length: 16
  address_family: IPv4
```

代理会处理该策略和其他信息，并将其转换为特定平台的 ipset 和 iptables 规则：

```

ipset rule:
Name: ta_f7b05c30ffa338fc063081060bf3
Type: hash:net
Header: family inet hashsize 1024 maxelem 65536
Size in memory: 16784
References: 1
Members:
1.2.0.0/16
Name: ta_1b97bc50b3374829e11a3e020859
Type: hash:net
Header: family inet hashsize 1024 maxelem 65536
Size in memory: 16784
References: 1
Members:
3.4.0.0/16
iptables rule:
TA_INPUT -p tcp -m set --match-set ta_f7b05c30ffa338fc063081060bf3 src -m set --match-
→set ta_1b97bc50b3374829e11a3e020859 dst -m multiport --sports 20:30 -m multiport --
→dports 40:50 -j ACCEPT

```

警告

ipset 内核模块

当在代理配置文件中启用执行并禁用保留规则时，在 Linux 主机上运行的代理会确保 ipset 内核模块有足够大的 `max_sets` 配置。如需更改，代理会使用新的 `max_sets` 值来重新加载 ipset 内核模块。如果启用了保留规则，代理会检查当前 ipset 模块 `max_sets` 值，但不会进行任何更改。当前配置的 `max_sets` 值可在 `cat /sys/module/ip_set/parameters/max_sets` 中找到。

主机防火墙备份

首次在代理配置文件中启用执行时，Linux 主机上运行的代理会在控制主机防火墙之前将 ipset 和 ip[6]tables 的当前内容存储在 `/opt/cisco/tetration/backup` 中。

执行配置的连续禁用或启用转换不会生成备份。在卸载代理后不会删除该目录。

WAF 模式下 Windows 平台上的代理执行

在 Windows 平台上，Cisco Secure Workload 代理会使用 Windows 防火墙执行网络策略。

高级安全 Windows 防火墙

Windows 上的本地组件（高级安全 Windows 防火墙）根据以下类型的设置来调节网络流量：

- 监管入站网络流量的规则。
- 监管出站网络流量的规则。
- 根据网络流量源和目标的身份验证状态来覆盖规则。
- 适用于 IPsec 流量和 Windows 服务的规则。

使用入站和出站防火墙规则对 Cisco Secure Workload 网络策略进行编程。

Cisco Secure Workload 规则和 Windows 防火墙

在 Windows 平台上，Cisco Secure Workload 网络策略按如下方式执行：

1. Cisco Secure Workload 网络策略中独立于平台的防火墙规则将转换为 Windows 防火墙规则。
2. 规则将在 Windows 防火墙中编程。
3. Windows 防火墙执行规则。
4. 系统将监控 Windows 防火墙及其规则集。如果检测到更改，系统会报告偏差，并在 Windows 防火墙中重置 Cisco Secure Workload Network 策略。

安全配置文件

Windows 防火墙会根据主机连接的网络对规则进行分组。这些规则组被称为配置文件，共有三个此类配置文件：

- 域配置文件
- 专用配置文件
- 公共配置文件

Cisco Secure Workload 规则会被编程到所有配置文件中，但只会持续监控活动配置文件中的规则。

有效设置和混合列表策略

Windows 防火墙中的规则集未根据优先级进行排序。当多个规则与数据包匹配时，最严格的规则将生效，这意味着 DENY 规则优先于 ALLOW 规则。有关详细信息，请参阅 [Microsoft TechNet](#) 上的文章。

请考虑“执行代理”部分中的混合列表（允许和拒绝）策略示例：

```
1. ALLOW 1.2.3.30 tcp port 80
2. ALLOW 1.2.3.40 udp port 53
3. BLOCK 1.2.3.0/24 ip
4. ALLOW 1.2.0.0/16 ip
5. Catch-all: DROP ingress, ALLOW egress
```

当流向主机 1.2.3.30 TCP 端口 80 的数据包到达防火墙时，它会匹配所有规则，但其中最严格的规则 3 将被执行，数据包将被丢弃。这种行为违背了按顺序评估规则、执行规则 1 以及允许数据包的预期。

由于上述 Windows 防火墙的设计，这种行为差异在 Windows 平台上是意料之中的。在具有不同规则操作的重叠规则的混合列表策略中，可以观察到这种行为。

例如，

```
1. ALLOW 1.2.3.30 tcp
2. BLOCK 1.2.3.0/24 tcp
```

来自其他防火墙或策略的干扰

建议您授予代理对 Windows 防火墙的完全和排他性控制，以便按预期执行 Cisco Secure Workload 网络策略。在以下情况时，代理无法可靠地执行策略：

- 存在第三方防火墙。（Windows 防火墙必须是主机上的活动防火墙产品。）
- 为当前配置文件禁用了防火墙。
- 使用组策略部署冲突的防火墙设置。一些冲突的设置包括：
 - 防火墙规则
 - 当前配置文件中与策略的捕获全部规则不同的默认入站或出站操作。
 - 为当前配置文件禁用了防火墙。

有状态的执行

Windows 高级防火墙被视为有状态的防火墙，即对于某些协议（例如 TCP），防火墙维护内部状态跟踪，以检测到达防火墙的新数据包是否属于已知连接。属于已知连接的数据包无需检查防火墙规则即可被允许。状态防火墙支持双向通信，而无需在 INBOUND 和 OUTBOUND 表中建立规则。

例如，请考虑适用于 Web 服务器的以下规则：**Accept all TCP connections to port 443**

这样做的目的是接受端口 443 上到服务器的所有 TCP 连接，并允许服务器与客户端进行通信。在这种情况下，INBOUND 表中只会插入一条规则，从而允许端口 443 上的 TCP 连接。无需在 OUTBOUND 表中插入任何规则。在 OUTBOUND 表中插入规则由 Windows 高级防火墙隐式执行。



Note 状态跟踪只适用于建立和维护显式连接的协议。对于其他协议，必须对 INBOUND 和 OUTBOUND 规则进行编程，以启用双向通信。

启用执行后，当协议为 TCP 时，给定的具体规则将被编程为有状态（代理会根据情景决定将该规则插入 INBOUND 表还是 OUTBOUND 表）。对于其他协议（包括 ANY），会同时编程 INBOUND 和 OUTBOUND 规则。

警告

主机防火墙备份

首次在代理配置文件中启用执行时，在 Windows 主机上运行的代理在控制主机防火墙之前，会将当前的 Windows 高级防火墙内容导出到 ProgramData\Cisco\Tetration\backup。执行配置连续禁用或启用转换不会生成备份。代理卸载时不会删除该目录。

WFP 模式下 Windows 平台上的代理执行

在 Windows 平台上，代理通过设置 Windows 过滤平台（WFP）过滤器来执行网络策略。不使用 Windows 高级防火墙配置网络策略。

Windows 过滤平台

Windows 过滤平台 (WFP) 是 Microsoft 提供的一组 API，用于配置过滤器以处理网络流量。网络流量处理过滤器使用内核级 API 和用户级 API 进行配置。WFP 过滤器可在各个层进行配置：网络层、传输层、应用层执行 (ALE)。Cisco Secure Workload WFP 过滤器在 ALE 层配置，类似于 Windows 防火墙规则。每个层都有多个子层，按权重从高到低排序。在每个子层中，过滤器按权重从高到低排序。网络数据包会遍历所有子层。在每个子层，网络数据包都会根据权重从高到低穿过匹配过滤器，并返回相应的操作：允许或阻止。通过所有子层后，数据包将根据“阻止”操作优先于“允许”操作的规则进行处理。

WFP 相对于 WAF 的优势

- 避免 Windows 防火墙配置依赖关系。
- 克服 GPO 限制。
- 确保轻松迁移和策略恢复。
- 允许您控制策略排序。
- 避免 Windows 防火墙的严格阻止优先策略顺序。
- 减少策略更新时的 CPU 开销。
- 创建高效的 1:1 策略规则过滤器。
- 确保更快的单步更新。

WFP 的代理支持

当执行配置为使用 WFP 时，Cisco Secure Workload 过滤器会覆盖 Windows 防火墙规则。

在 WFP 模式下，代理配置以下 WFP 对象：

- 提供者具有 GUID 和名称，用于过滤器管理，不会影响数据包过滤
- 子层具有 GUID、名称和权重。Cisco Secure Workload 子层配置的权重高于 Windows 高级防火墙子层。
- 过滤器具有名称、GUID、ID、权重、层 ID、子层密钥、操作 (PERMIT/BLOCK) 和条件。为 Golder 规则、自身规则和策略规则配置 WFP 过滤器。代理还会配置端口扫描防护过滤器。Cisco Secure Workload 过滤器配置了 FWPM_FILTER_FLAG_CLEAR_ACTION_RIGHT 标志。此标志可确保 Cisco Secure Workload 过滤器不会被 Microsoft 防火墙规则覆盖。对于每个 Cisco Secure Workload 网络策略规则，系统会根据方向（入站或出站）和协议来配置一个或多个 WFP 过滤器。

对于 TCP 入站策略，

```
id: 14 , TCP Allow 10.195.210.184 Dir=In localport=3389
```

配置的 WFP 过滤器如下：

```

Filter Name:                Cisco Secure Workload Rule 14
-----
EffectiveWeight:           18446744073709551589
LayerKey:                  FWPM_LAYER_ALE_AUTH_LISTEN_V4
Action:                    Permit
Local Port:                3389
Filter Name:                Cisco Secure Workload Rule 14
-----
EffectiveWeight:           18446744073709551589
LayerKey:                  FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V4
Action:                    Permit
RemoteIP:                  10.195.210.184-10.195.210.184

```

Cisco Secure Workload 代理分别为入站和出站捕获全部策略配置 **Cisco Secure Workload** 默认入站和 **Cisco Secure Workload** 默认出站过滤器。

代理 WFP 支持和 Windows 防火墙

- 代理不监控 WAF 规则或 WAF 配置文件。
- 代理不监控防火墙状态。
- 代理不需要启用防火墙状态。
- 代理不与 GPO 策略冲突。

有效设置和混合列表策略

WFP 模式下的代理执行支持混合列表或灰名单策略。

请考虑“执行代理”部分中的混合列表（允许和拒绝）策略示例：

```

1. ALLOW 1.2.3.30 tcp port 80-          wt1000
2. BLOCK 1.2.3.0/24 ip-                 wt998
3. ALLOW 1.2.0.0/16 ip-                 wt997
4. Catch-all: DROP ingress, ALLOW egress - wt996

```

当发向主机 1.2.3.30 tcp 端口 80 的数据包到达防火墙时，它会匹配过滤器 1 并被允许。但由于过滤器 2，流向主机 1.2.3.10 的数据包会被阻止。过滤器 3 允许发往主机 1.2.2.10 的数据包。

有状态的执行

Cisco Secure Workload WFP 过滤器在 ALE 层进行配置。网络流量会针对套接字 connect()、listen() 和 accept() 操作进行过滤。建立连接后，与 L4 连接相关的网络数据包不会被过滤。

已配置的 WFP 过滤器的可视性

您可以使用 `c:\program files\tetration\tetenf.exe` 来查看已配置的 Cisco Secure Workload WFP 过滤器。支持的选项：

- 使用管理权限运行 `cmd.exe`。
- 运行 `c:\program files\tetration\tetenf.exe -l -f <-verbose> <-output=outfile.txt>`。

或

- 使用管理权限运行 `cmd.exe`。
- 运行 `netsh wfp show filters`。
- 查看 `filters.xml` 文件以获取已配置的 Cisco Secure Workload 过滤器。

在 WFP 模式下禁用隐身模式过滤器

要禁用隐身模式过滤器（端口扫描过滤器），请执行以下操作：

过程

步骤 1 编辑 `\conf\enforcer.cfg`。

步骤 2 添加 `disable_wfp_stealth_mode: 1`

步骤 3 保存文件。

步骤 4 使用管理权限，通过以下方式重启 `tetenforcer` 服务：

- a) 运行命令：`sc stop tetenforcer` 以停止 TetEnforcer 服务。
- b) 运行命令：`sc start tetenforcer` 以启动 TetEnforcer 服务。

步骤 5 验证：

- a) 使用管理权限运行 `cmd.exe`。
- b) 运行命令：`c:\program files\tetration\tetenf.exe -l -f <-verbose> <-output=outfile.txt>`。

“Tetration Internal Rule block portscan” filters are not configured.

删除已配置的 WFP 过滤器

您可以使用 `c:\program files\tetration\tetenf.exe` 来删除已配置的 Cisco Secure Workload WFP 过滤器。为避免意外删除过滤器，当您运行 `delete` 命令时，请以 `<yyyymm>` 格式指定令牌，其中 `yyyy` 是当前年份，`mm` 是当前月份的数字形式。例如，如果今天的日期是 2021/01/21，则令牌为 **-token=202101**

支持的选项包括：

- 使用管理权限运行 `cmd.exe`。
- 要删除所有已配置的 Cisco Secure Workload 过滤器，请运行 `c:\program files\tetration\tetenf.exe -d -f -all -token=<yyyymm>`
- 要删除所有已配置的 Cisco Secure Workload WFP 对象，请运行 `c:\program files\tetration\tetenf.exe -d -all -token=<yyyymm>`
- 要按名称删除 Cisco Secure Workload WFP 过滤器，请运行 `c:\program files\tetration\tetenf.exe -d -name=<WFP filter name> -token=<yyyymm>`

WFP 模式下的已知限制

- 将“执行模式”(Enforcement Mode) 设置为 WFP 时，代理配置文件中的保留规则 (**Preserve Rules**) 设置将不起作用。

为 Windows 属性配置策略

要在基于 Windows 的工作负载上执行策略时更精细地执行策略，则可以通过以下方式过滤网络流量：

- 应用名称
- 服务名称
- 带或不带用户组的用户名

WAF 和 WFP 模式均支持此选项。基于 Windows 操作系统的过滤器在生成的网络策略中分为使用者过滤器和提供者过滤器。使用者过滤器会过滤在使用者工作负载上发起的网络流量，而提供者过滤器会过滤发往提供者工作负载的网络流量。

开始之前

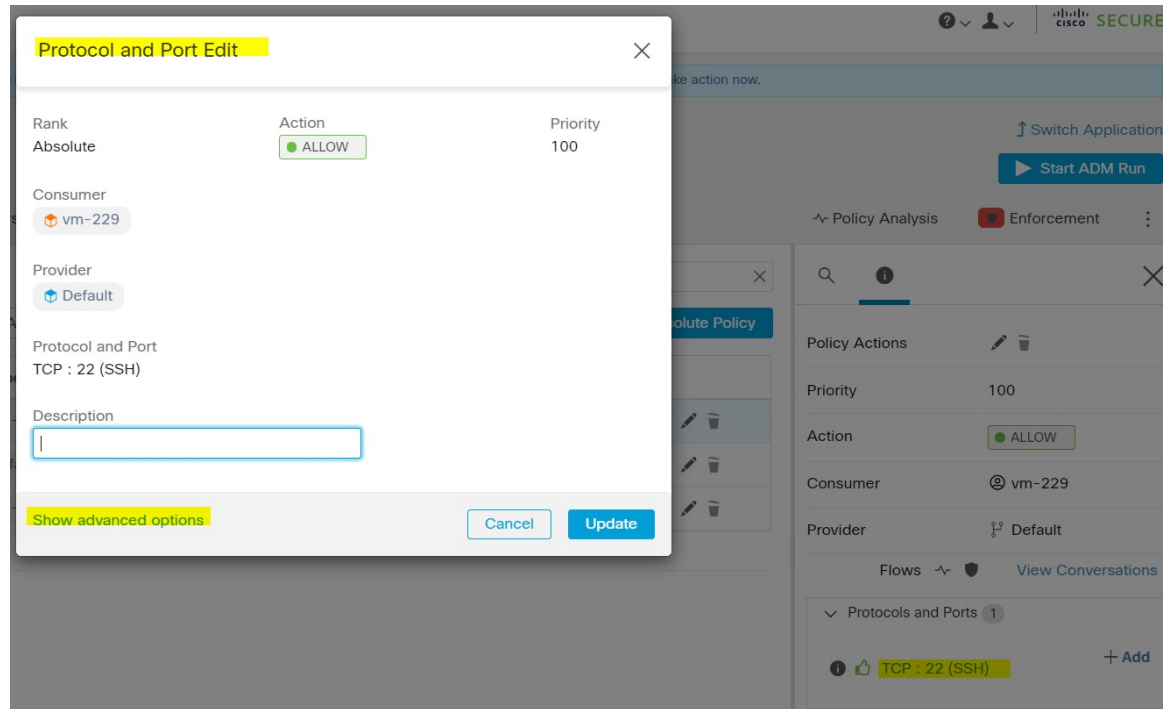
此程序假定您要修改现有的策略。如果尚未创建要添加基于 Windows 操作系统的过滤器的策略，请先创建该策略。



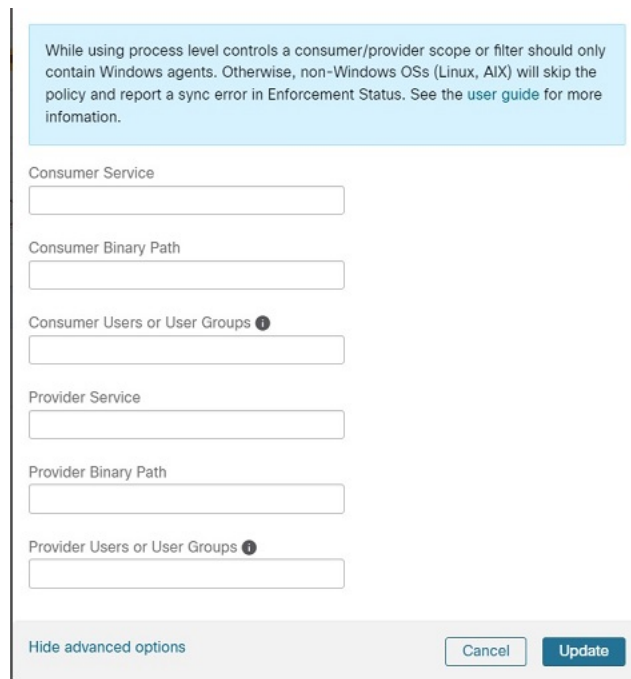
重要事项 有关涉及 Windows 属性的策略，请参阅[警告，第 43 页](#)和[已知限制，第 43 页](#)。

过程

- 步骤 1** 在导航窗格中，点击**防御 (Defend) > 分段 (Segmentation)**。
- 步骤 2** 点击包含要为其配置基于 Windows 操作系统的过滤器的策略的范围。
- 步骤 3** 点击要在其中编辑策略的工作空间。
- 步骤 4** 点击**管理策略 (Manage Policies)**。
- 步骤 5** 选择要编辑的策略。
重要事 使用者和提供者必须仅包含 Windows 工作负载。
项
- 步骤 6** 在要编辑的策略的表行中，点击**协议和端口 (Protocols and Ports)** 列中的现有值。
- 步骤 7** 在右侧窗格中，点击**协议和端口 (Protocols and Ports)** 下的现有值。
在本示例中，点击 **TCP : 22 (SSH)**。



步骤 8 点击显示高级选项 (Show advanced options)。



步骤 9 根据应用名称、服务名称或用户名配置使用者过滤器。

- 应用名称必须是完整路径名。
- 服务名称必须是短服务名称。

- 用户名可以是本地用户名（例如，`tetter`）或域用户名（例如，`sensor-dev@sensor-dev.com` 或 `sensor-dev\sensor-dev`）
- 用户组可以是本地用户组（例如，`Administrators`）或域用户组（例如，`domain users\sensor-dev`）
- 可以指定多个用户名和/或用户组名并以“,”分隔。（例如，`sensor-dev\@sensor-dev.com,domain users\sensor-dev`）
- 服务名称和用户名不能同时配置。

步骤 10 根据应用名称、服务名称或用户名配置提供者过滤器。

遵循上一步中为使用者过滤器提供的相同准则。

步骤 11 输入二进制文件的路径（如适用）。

例如，输入 `c:\test\putty.exe`

步骤 12 点击更新 (Update)。

建议的基于 Windows 操作系统的策略配置

尽可能在策略中指定端口和协议；建议不要允许任何端口、任何协议。

例如，生成的具有端口和协议限制的策略可能如下所示：

```
dst_ports {
  start_port: 22
  end_port: 22
  consumer_filters {
    application_name: "c:\\test\\putty.exe"
  }
}
ip_protocol: TCP
```

相反，如果您允许 `iperf.exe` 使用任何协议和任何端口发起的网络连接，则生成的策略将如下所示：

```
match_set {
  dst_ports {
    end_port: 65535
    consumer_filters {
      application_name: "c:\\test\\iperf.exe"
    }
  }
  address_family: IPv4
  inspection_point: EGRESS
  match_comment: "PolicyId=61008290755f027a92291b9d:61005f90497d4f47cedacb86:"
}
```

对于上述过滤器，Cisco Secure Workload 创建一条策略规则以允许提供者上的网络流量，如下所示：

```
match_set {
  dst_ports {
```

```
        end_port: 65535
      }
      address_family: IPv4
      inspection_point: INGRESS
      match_comment: "PolicyId=61008290755f027a92291b9d:61005f90497d4f47cedacb86:"
    }
  }
```

此网络规则会打开提供者上的所有端口。我们强烈建议不要使用 *Any* 协议来创建基于操作系统的过滤器。

已知限制

- Windows 2008 R2 不支持基于 Windows 操作系统的过滤策略。
- 可以使用单个用户名来配置网络策略，而 MS 防火墙 UI 支持多个用户。

警告

- 在使用基于 Windows 操作系统的策略时，使用者/提供者范围或过滤器应仅包含 Windows 代理。否则，非 Windows 操作系统（Linux、AIX）会跳过策略并在“执行状态” (Enforcement Status) 中报告同步错误。
- 避免创建过滤条件宽松的 Windows 操作系统过滤器。此类条件可能会打开不需要的网络端口。
- 如果操作系统过滤器是为使用者配置的，那么策略只适用于使用者，同样，如果是为提供者配置的，那么策略只适用于提供者。
- 由于对网络流的进程上下文、用户上下文或服务上下文了解有限或一无所知，如果策略采用基于 Windows 操作系统的过滤器，策略分析就会出现偏差。

使用基于 Windows 操作系统的过滤属性对策略进行验证和故障排除

如果使用基于 Windows 操作系统的过滤属性，则以下主题将为您提供验证和故障排除信息。

思科 TAC 可以根据需要使用这些信息来对此类策略进行故障排除。

基于应用名称的策略

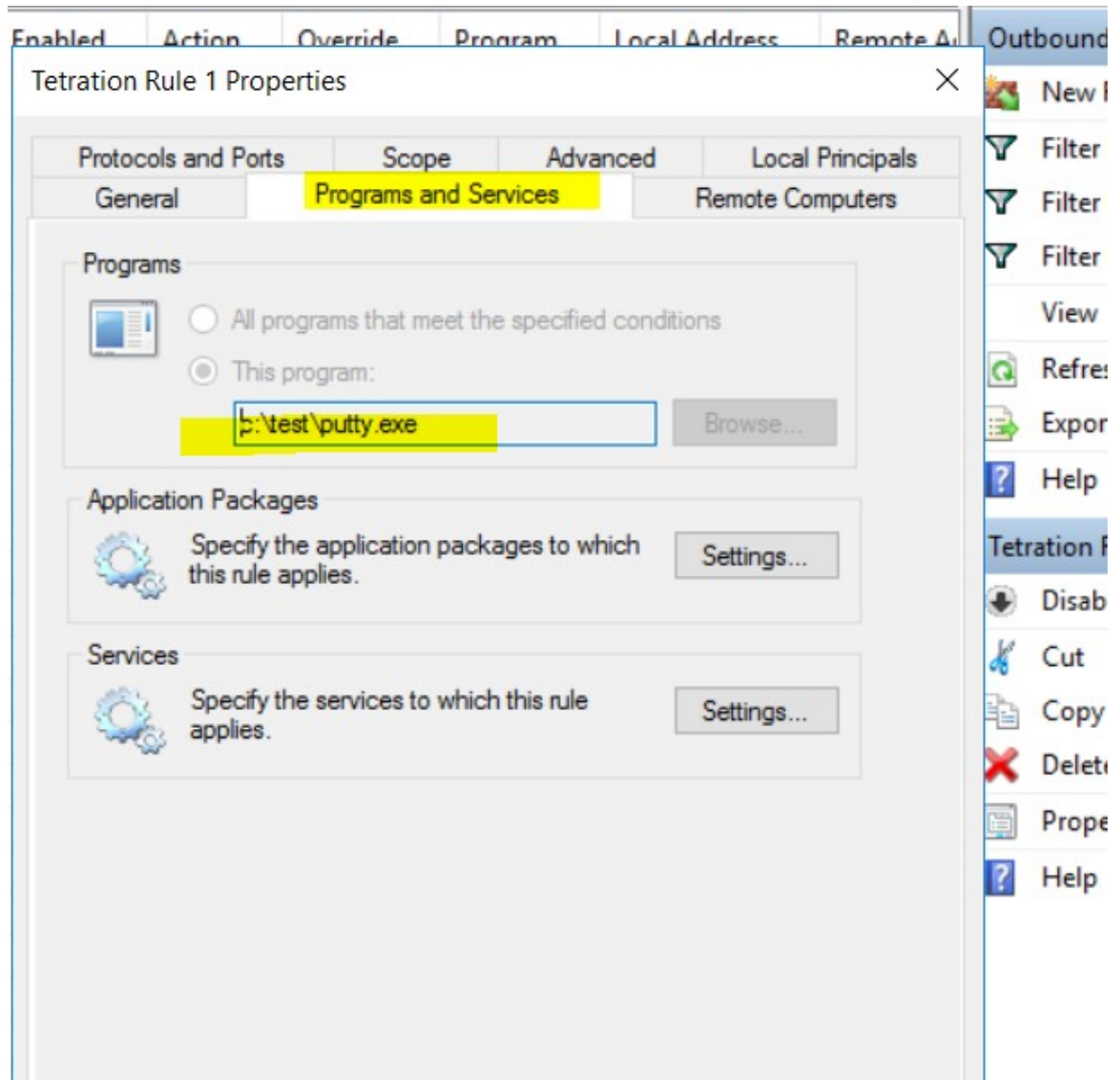
使用以下信息验证 Windows 操作系统工作负载上基于应用名称的策略并排除故障。

以下部分介绍策略在输入为 `c:\test\putty.exe` 的应用二进制文件的工作负载上应如何显示。

基于应用名称的示例策略

```
dst_ports {
  start_port: 22
  end_port: 22
  consumer_filters {
    application_name: "c:\test\putty.exe"
  }
}
ip_protocol: TCP
address_family: IPv4
inspection_point: EGRESS
```

生成的防火墙规则



使用 netsh 生成的过滤器

要使用本地 Windows 工具验证过滤器是否已添加到高级策略中，请执行以下操作：

- 使用管理权限运行 `cmd.exe`。
- 运行 `netsh wfp show filters`。
- 在当前目录中生成输出文件 **filters.xml**。
- 检查 `FWPM_CONDITION_ALE_APP_ID` 以获取输出文件中的应用名称：`filters.xml`。

```
<fieldKey>FWPM_CONDITION_ALE_APP_ID</fieldKey>
      <matchType>FWP_MATCH_EQUAL</matchType>
      <conditionValue>
```

```

        <type>FWP_BYTE_BLOB_TYPE</type>
        <byteBlob>
            <data>
→5c006400650076006900630065005c0068006100720064006400690073006b0076006f006
→</data>
            <asString>\device\harddiskvolume2\temp\putty.exe</
→asString>
        </byteBlob>
    </conditionValue>

```

使用 `tetenf.exe -l -f` 生成的 WFP 过滤器

```

Filter Name:                Cisco Secure Workload Rule 1
-----
EffectiveWeight:           18446744073709551592
LayerKey:                  FWPM_LAYER_ALE_AUTH_CONNECT_V4
Action:                    Permit
RemoteIP:                  10.195.210.15-10.195.210.15
Remote Port:               22
Protocol:                  6
AppID:                     \device\harddiskvolume2\test\putty.exe

```

无效的应用名称

- 在 WAF 模式下，为无效的应用名称创建了防火墙规则。
- 在 WFP 模式下，不会为无效的应用名称创建 WFP 过滤器，但不会拒绝 NPC。代理会记录警告消息并配置其余策略规则。

基于服务名称的策略

使用以下信息会根据 Windows 操作系统工作负载上的服务名称来验证策略并进行故障排除。

以下各节介绍策略应在工作负载上显示的方式。

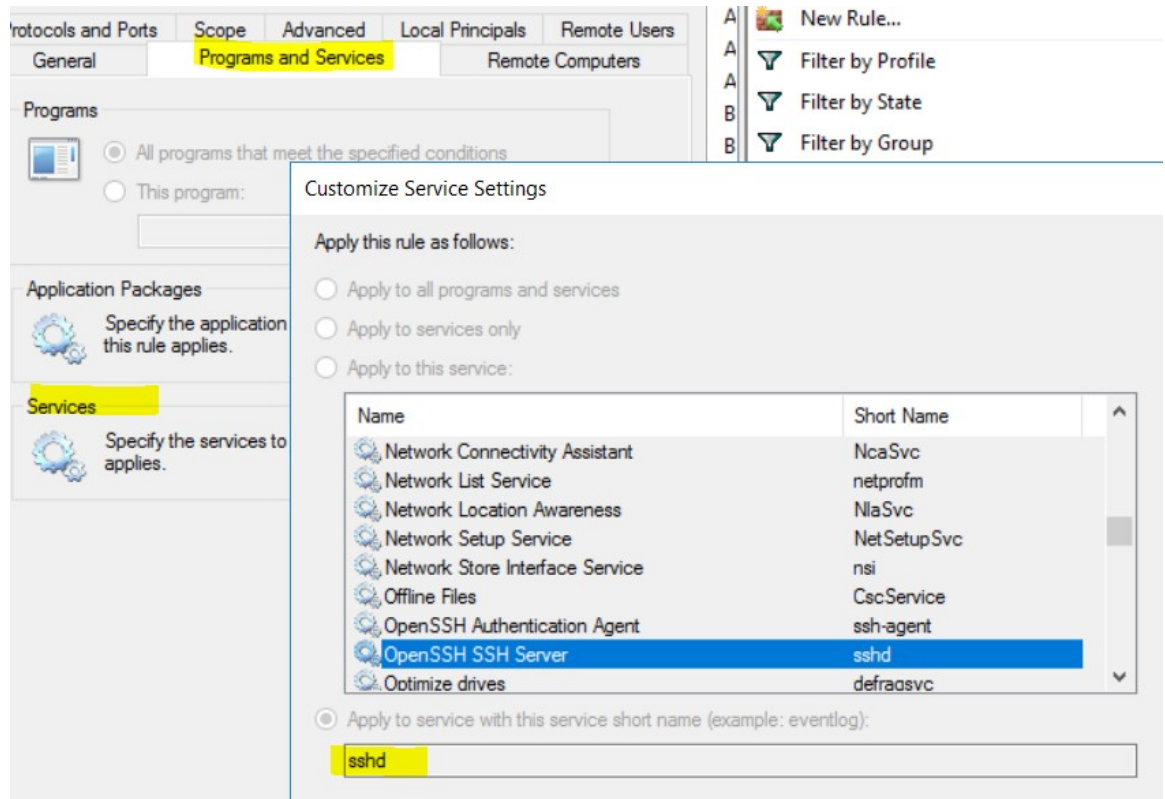
基于服务名称的示例策略

```

dst_ports {
    start_port: 22
    end_port: 22
    provider_filters {
        service_name: "sshd"
    }
}
ip_protocol: TCP
address_family: IPv4
inspection_point: INGRESS

```

生成的防火墙规则



使用 netsh 生成的过滤器

要使用本地 Windows 工具验证是否已为高级策略添加过滤器，请执行以下操作：

- 使用管理权限运行 `cmd.exe`。
- 运行 `netsh wfp show filters`。
- 在当前目录中生成输出文件 `filters.xml`。
- 检查 `FWPM_CONDITION_ALE_USER_ID` 以获取输出文件 `filters.xml` 中的用户名。

```
<item>
    <fieldKey>FWPM_CONDITION_ALE_USER_ID</fieldKey>
    <matchType>FWP_MATCH_EQUAL</matchType>
    <conditionValue>
        <type>FWP_SECURITY_DESCRIPTOR_TYPE</type>
        <sd>0:SYG:SYD: (A;;CCRC;;;S-1-5-80-3847866527-469524349-687026318-
        →516638107)</sd>
    </conditionValue>
</item>
```

使用 `tetenf.exe -l -f` 生成的 WFP 过滤器

```
Filter Name:      Cisco Secure Workload Rule 3
-----
```

```
EffectiveWeight:      18446744073709551590
LayerKey:             FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V4
Action:              Permit
Local Port:          22
Protocol:            6
User or Service:     NT SERVICE\sshd
```

无效的服务名称

- 在 WAF 模式下，系统会为不存在的服务名称创建防火墙规则。
- 在 WFP 模式下，系统不会为不存在的服务名称创建 WFP 过滤器。
- 服务 SID 类型必须为不受限制 (*Unrestricted*) 或受限制 (*Restricted*)。如果服务类型为无 (*None*)，则可以添加防火墙规则和 WFP 过滤器，但它们不起作用。

要验证 SID 类型，请运行以下命令：

```
sc qsidtype <service name>
```

基于用户组或用户名的策略

使用以下信息会根据 Windows 操作系统工作负载上的用户名（带和不带用户组名称）来验证策略并进行故障排除。

本主题中的部分介绍策略应在工作负载上显示的方式。

本主题中的示例基于使用以下信息配置的策略：

Figure 2: 基于用户组或用户名的策略

Description

While using process level controls, a consumer/provider scope or filter should only contain Windows agents. Otherwise, non-Windows OSs (Linux, AIX) will skip the policy and report a sync error in Enforcement Status. See the user guide for more information.

Consumer Service

Consumer Binary Path

Consumer Users or User Groups ⓘ
sensor-dev\domain users,sensor-dev@se

Provider Service

Provider Binary Path

Provider Users or User Groups ⓘ

基于用户名的策略示例

```
dst_ports {
  start_port: 30000
  end_port: 30000
  provider_filters {
    user_name: "sensor-dev\sensor-dev"
  }
}
ip_protocol: TCP
address_family: IPv4
inspection_point: EGRESS
```

基于用户组和用户名的策略示例

```
dst_ports {
  start_port: 30000
  end_port: 30000
  provider_filters {
    user_name: "sensor-dev\domain users,sensor-dev\sensor-dev"
  }
}
ip_protocol: TCP
```

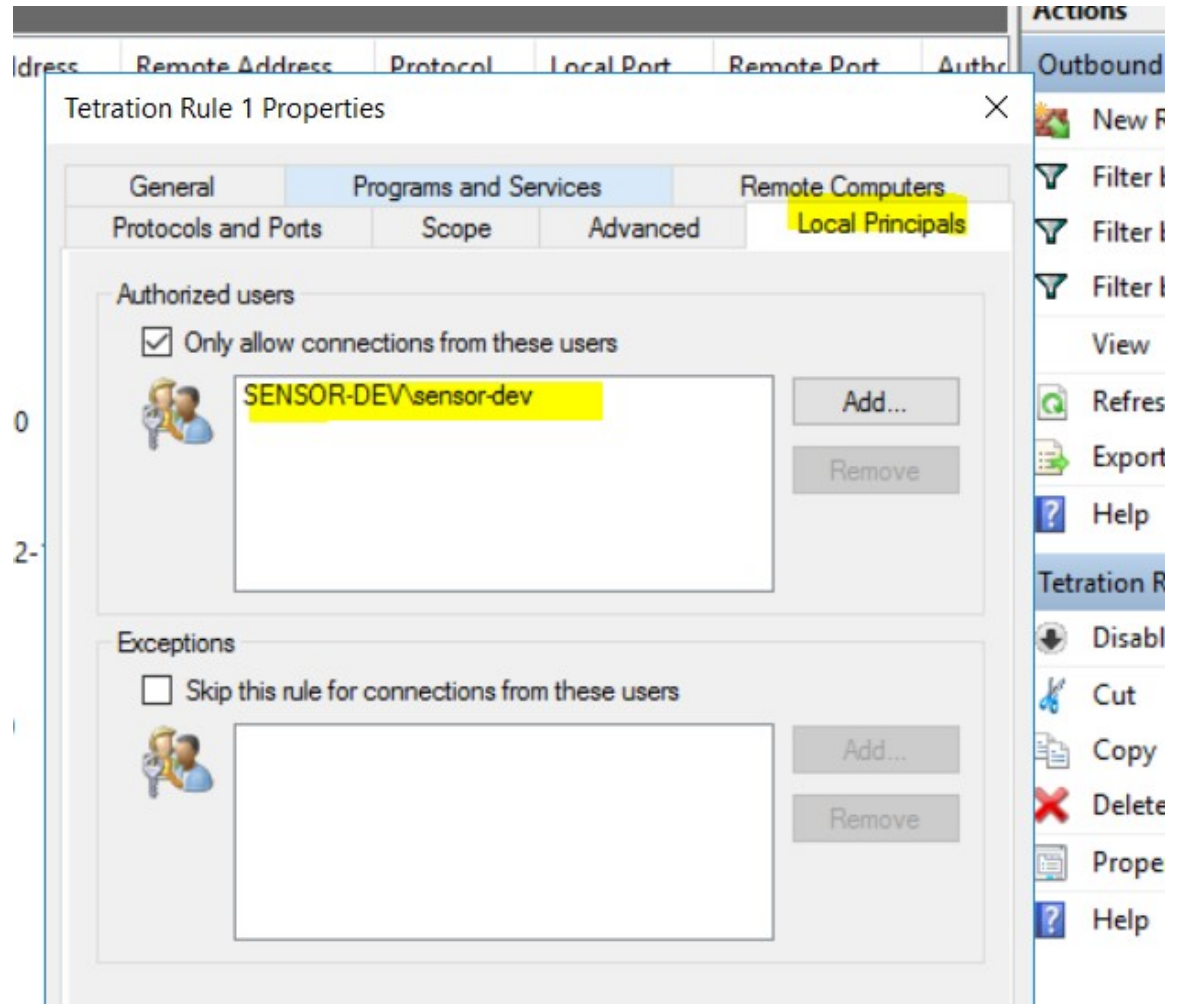


```
address_family: IPv4
inspection_point: EGRESS
```

生成的防火墙规则

基于用户名的防火墙规则

示例：基于用户名 sensor-dev\sensor-dev 的防火墙规则



基于用户组和用户名的防火墙规则

示例：基于用户名 sensor-dev\sensor-dev 和用户组 domain users\sensor-dev 的防火墙规则


```

        <type>FWP_SECURITY_DESCRIPTOR_TYPE</type>
        <sd>O:LSID: (A;;CC;;;S-1-5-21-4172447896-825920244-2358685150) </sd>
    </conditionValue>
</item>

```

使用 `tetenf.exe -l -f` 生成的 WFP 过滤器

基于用户名过滤

示例：基于用户名 `SENSOR-DEV\sensor-dev` 的 WFP 规则

```

Filter Name:                Cisco Secure Workload Rule 1
-----
EffectiveWeight:           18446744073709551590
LayerKey:                  FWPM_LAYER_ALE_AUTH_CONNECT_V4
Action:                    Permit
RemoteIP:                  10.195.210.15-10.195.210.15
Remote Port:               30000
Protocol:                  6
User or Service:           SENSOR-DEV\sensor-dev

```

基于用户组 and 用户名过滤

示例：基于用户名 `SENSOR-DEV\sensor-dev` 和用户组名 `SENSOR-DEV\Domain Users` 的 WFP 规则

```

Filter Name:                Cisco Secure Workload Rule 1
-----
EffectiveWeight:           18446744073709551590
LayerKey:                  FWPM_LAYER_ALE_AUTH_CONNECT_V4
Action:                    Permit
RemoteIP:                  10.195.210.15-10.195.210.15
Remote Port:               30000
Protocol:                  6
User or Service:           SENSOR-DEV\Domain Users, SENSOR-DEV\sensor-dev

```

无法为网络策略规则配置服务名称和用户名。



Note 如果用户名或用户组无效，则网络策略会被 Windows 代理拒绝。

在 Windows 节点上执行 Kubernetes Pod

在 Windows 工作节点上安装 Kubernetes 守护进程集代理后，它会在 AKS 环境中捕获来自 Windows 工作节点和 Kubernetes Pod 的网络流。

要求

- 在具有 Windows 节点的 AKS 环境中支持执行 Kubernetes Pod。
- 执行模式必须为 WFP，并将保留规则 (**Preserve Rules**) 设置为“关” (Off)。
- 支持 Microsoft Windows Server 2019 和 Windows Server 2022。

这些策略在 vSwitch 上针对使用 VFP 连接到 Pod 的端口执行。虚拟过滤平台 (VFP) 是 vSwitch 的一个组件，用于配置过滤器以处理网络流量。执行策略时，“保留模式” (Preserve Mode) 处于关闭状态。

每个过滤器具有以下属性：

- Id: 过滤器名称
- Direction: 传入或传出
- RuleType: 交换机或主机。
 - 当类型为“交换机”时，则在 vSwitch 上配置过滤器。
 - 当类型为“主机”时，则创建 WFP 过滤器。
- Action: 允许或阻止
- LocalPorts: 这可以是端口或范围。例如，80 或 100-200。
- RemotePorts: 与本地端口相同。
- LocalAddresses: 它是地址或范围。例如，10.224.0.5、10.224.1.0/24（不允许使用 10.224.1.1-10.224.1.10）。
- RemoteAddress: 与本地地址相同
- Protocol: ICMP/TCP/UDP/IGMP 协议 255 为 IPPROTO_RAW，256 - PROTO_MAX

只能为 UDP 和 TCP 指定端口，并且除非指定了协议，否则策略中不允许使用端口。

在虚拟端口上配置策略是一项基于事务的操作。如果其中一个过滤器无效，则整个策略的执行就会失败。

这是有状态的执行。当前不支持基于应用、用户或服务的策略。

与 Calico 的兼容性

Pod 执行在“保留规则”关闭模式下工作。当 Windows 代理在 Pod 上执行规则时，它会删除已配置的策略。如果 Calico 插件在代理之后执行网络策略，则代理将其识别为偏差，并删除由 Calico 配置的网络策略，并重新执行代理策略。



Note 在 Windows 节点上卸载 Windows 代理时，将删除已执行的策略。

已配置 VFP 过滤器的可视性

使用 Cisco Secure Workload 列出 Pod 过滤器的选项不可用。在 AKS 环境中，可以使用内置 PowerShell 脚本。运行以下 PowerShell 脚本：c:\k\debug\collectlogs.ps1。查看已配置过滤器的输出文件 **vfoutput.txt** 和 **hnsdiag.txt**。

删除 Windows 代理配置的 VFP 过滤器

1. 使用管理权限运行 **cmd.exe**。
2. 运行命令：<installation folder>\tetenf.exe -d -f -pods -token=<yyyyymm>。



Note 命令会删除所有 Pod 的 VFP 过滤器。

对执行的策略和网络流进行故障排除

1. 运行命令: `netsh wfp start capture keywords=19`。
2. 运行网络流量。
3. 停止捕获流: `netsh wfp stop capture`。
4. 从 `wfpdiag.cab` 文件中提取 `wfpdiag.xml`。查看已丢弃的流。

要将允许或丢弃的网络流映射到 Pod 策略, 请执行以下操作:

1. 启动 ETW 会话: `logman start <session name> -p Microsoft-Windows-Hyper-V-VfpExt -o <output file.etl> -ets`
2. 运行网络流量。
3. 停止捕获流: `logman stop <session name>`。
4. 在命令提示符中, 运行: `tracert <output file.etl>`。命令会创建 `dumpfile.xml` 文件。查看网络流。

AIX 平台上的代理执行

在 AIX 平台上, Cisco Secure Workload 代理会使用 IPFilter 实用程序来执行网络策略。默认情况下, 在主机上启用代理后, 代理会控制和编程 IPv4 过滤表。不支持 IPv6 执行。

IPFilter

AIX 上的 IPFilter 软件包用于提供防火墙服务, 并在 AIX 上作为内核扩展包提供。它作为内核扩展模块 `/usr/lib/drivers/ipf` 进行加载。它包括用于编程 ipfilter 规则的 `ipf`、`ippool`、`ipfstat`、`ipmon`、`ipfs` 和 `ipnat` 实用程序, 每个规则指定数据包的匹配条件。有关详细信息, 请参阅 AIX 手册中的 IPFilter 页面。

启用执行时, 代理使用 IPFilter 对包含允许或丢弃 IPv4 数据包的规则的 IPv4 过滤表进行编程。代理会将这些规则分组, 以使用控制器来对策略进行分类和管理。这些规则包括从策略派生的 Cisco Secure Workload 规则和代理生成的规则。

当代理收到独立于平台的规则时, 它会解析这些规则并将其转换为 `ipfilter` 或 `ippool` 规则, 并将这些规则插入过滤器表中。对防火墙进行编程后, 执行代理会监控防火墙是否存在任何规则或策略偏差, 如果是, 则重新编程防火墙。代理会跟踪防火墙中编程的策略, 并定期向控制器报告其状态。

独立于平台的网络策略消息中的典型策略包括:

```
source set id: "test-set-1"
destination set id: "test-set-2"
source ports: 20-30
```

```

destination ports: 40-50
ip protocol: TCP
action: ALLOW
...
set_id: "test-set-1"
  ip_addr: 1.2.0.0
  prefix_length: 16
  address_family: IPv4
set_id: "test-set-2"
  ip_addr: 5.6.0.0
  prefix_length: 16
  address_family: IPv4

```

代理会与其他信息一起处理策略，并将其转换为特定于平台的 `ippool` 和 `ipfilter` 规则：

```

table role = ipf type = tree number = 51400
{ 1.2.0.0/16; };

table role = ipf type = tree number = 75966
{ 5.6.0.0/16; };

pass in quick proto tcp from pool/51400 port 20:30 to pool/75966 port 40:50 flags S/SA group
  TA_INPUT
pass out quick proto tcp from pool/75966 port 40:50 to pool/51400 port 20:30 flags A/A group
  TA_OUTPUT

```

警告

主机防火墙备份

首次在代理配置文件中启用执行时，在 AIX 主机上运行的代理在控制主机防火墙之前，会将 `ippool` 和 `ipfilter` 的当前内容存储到 `/opt/cisco/tetration/backup` 中。执行配置连续禁用或启用转换不会生成备份。代理卸载时不会删除该目录。

已知限制

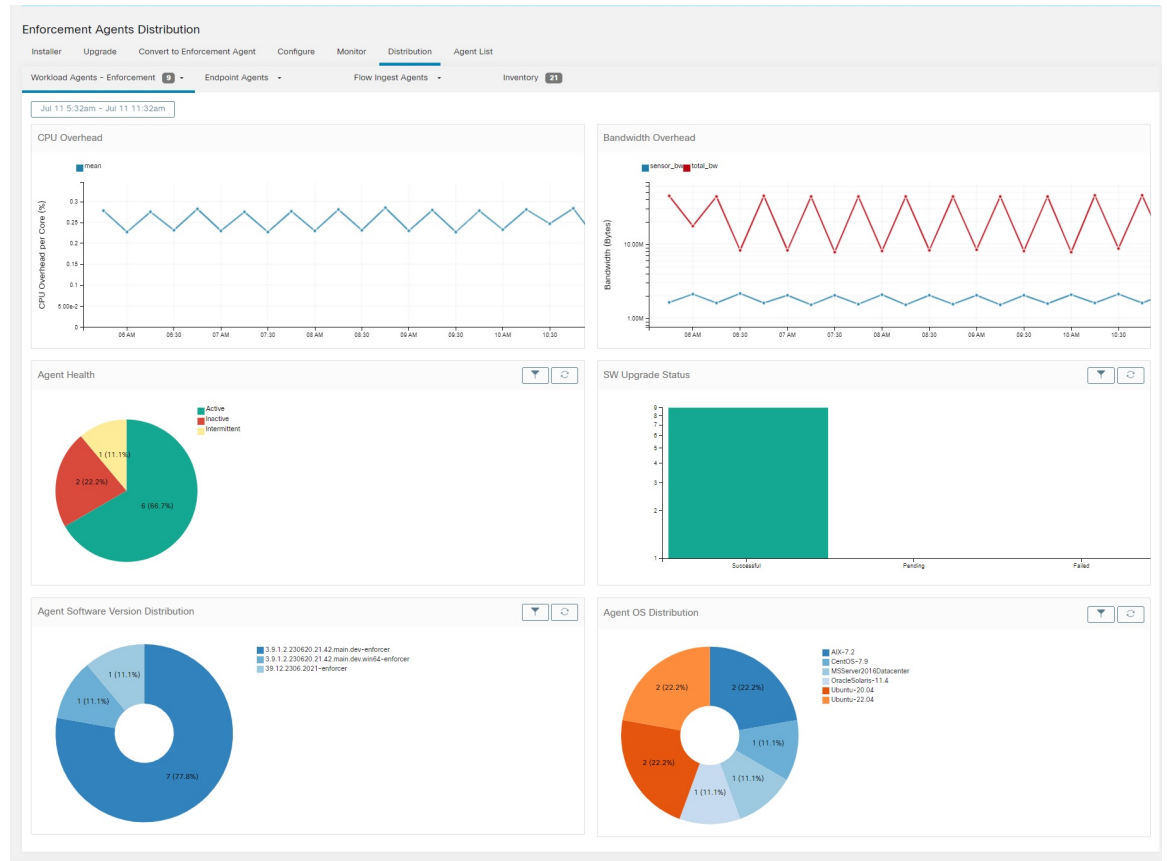
不支持 IPv6 执行。

检查代理状态和统计信息

Procedure

- 步骤 1** 从导航窗格中，依次选择**管理 (Manage)** > **工作负载 (Workloads)** > **代理 (Agents)**。
- 步骤 2** 点击**分布 (Distribution)** 选项卡。
- 步骤 3** 点击页面顶部的代理类型。
- 步骤 4** 在此页面上，您可以检查 CPU 开销、带宽开销、代理运行状况、软件更新状态、代理软件版本分布和代理操作系统分布。

Figure 3: “代理分布” (Agent Distribution) 页面



Note 代理运行状况：代理会每 10-30 分钟定期检查一次。如果超过 1 小时 30 分钟未签入，则代理处于非活动状态。为减少误报，如果签入间隔在 1 小时到 1 小时 30 分钟之间，代理运行状态将被设置为间歇性而非不活动状态。

有关执行状态的详细信息，请参阅“执行状态”部分。

查看代理详细信息

以下步骤提供用于导航至“工作负载配置文件” (Workload Profile) 页面的可用选项之一，该页面显示有关工作负载及其已安装代理的详细信息。

Procedure

步骤 1 在导航窗格中，点击整理 (Organize) >> 范围和资产 (Scopes and Inventory)。

步骤 2 搜索要查看其详细信息的工作负载。

步骤 3 点击 IP 地址以查看详细信息，例如代理运行状况、IP 地址、范围、资产类型、执行组、试验组、用户标签和流量（总字节数/总数据包数）。

有关详细信息，请参阅 [适用的工作负载](#)。

软件代理配置

配置软件代理的要求和前提条件

- 确保您具有所需的 Cisco Secure Workload 用户角色凭证：
 - 站点管理员
 - 客户支持

有关详细信息，请参阅 [用户角色和对代理配置的访问权限](#), on page 56。

- 确保您在主机上拥有在每一个工作负载上运行代理服务的权限。有关详细信息，请参阅 [代理的服务管理](#)。
- 验证代理的支持平台、要求和安装说明。有关详细信息，请参阅 [部署软件代理](#)。

用户角色和对代理配置的访问权限

1. 根范围所有者只有创建配置文件和配置意图规范的权限。
2. 作为根范围所有者，您可以创建只与所拥有的范围相关联的配置文件，并将这些配置文件强制用于代理。



Note 在“代理配置文件” (Agent Configuration Profile) 下，您现在可以在编辑配置文件之前使用配置文件查看意图数。

Figure 4: 范围所有者的软件代理配置

The screenshot displays the 'Agent Config Profiles' and 'Agent Config Intents' sections of the Cisco Secure Workload configuration interface.

Agent Config Profiles: A table with columns 'Name', 'Config', and 'Actions'. The 'Default' profile is selected. The 'Config' column lists various settings with status indicators (green dot for enabled, grey X for disabled):

- Enforcement
 - Windows Enforcement Mode - WFP (Enabled)
 - Presence Rules (Disabled)
 - Allow Broadcast (Enabled)
 - Allow Multicast (Enabled)
 - Allow Link Local Addresses (Enabled)
 - CPU Quota Mode - Adjusted (3%) (Enabled)
 - Memory Quota Limit - 512MB (Enabled)
- Flow Visibility
 - Flow Analysis Fidelity - Conversations (Enabled)
 - Data Plane (Enabled)
 - Auto-Upgrade (Enabled)
- Service Protection
 - PID Lookup (Disabled)
 - Service Protection (Enabled)
 - CPU Quota Mode - Adjusted (3%) (Enabled)
 - Memory Quota Limit - 512MB (Enabled)
 - Cleanup Period (Disabled)
 - Flows Disk Quota - 512MB (Enabled)
- Process Visibility and Forensics
 - Forensics (Disabled)
 - Process Visibility (Enabled)
 - Package Visibility (Enabled)
 - Meltdown Exploit Detection (Disabled)
 - CPU Quota Mode - Adjusted (3%) (Enabled)
 - Memory Quota Limit - 256MB (Enabled)

The 'Actions' column for the 'Default' profile includes an 'Edit' button and the text 'Used by 1 Intent'.

Agent Config Intents: A section with a 'Create Intent' button. It shows 'Apply profile Default to filter Everything'. Below this, there are sections for 'Interface Config Intents' (No intents found) and 'Agent Remote VRF Configurations' (No configs found), each with a 'Create Intent' or 'Create Config' button.

3. 站点管理员可以访问“代理配置”(Agent Configuration)页面中的所有组件，包括指定接口配置意图、远程虚拟路由和转发配置。

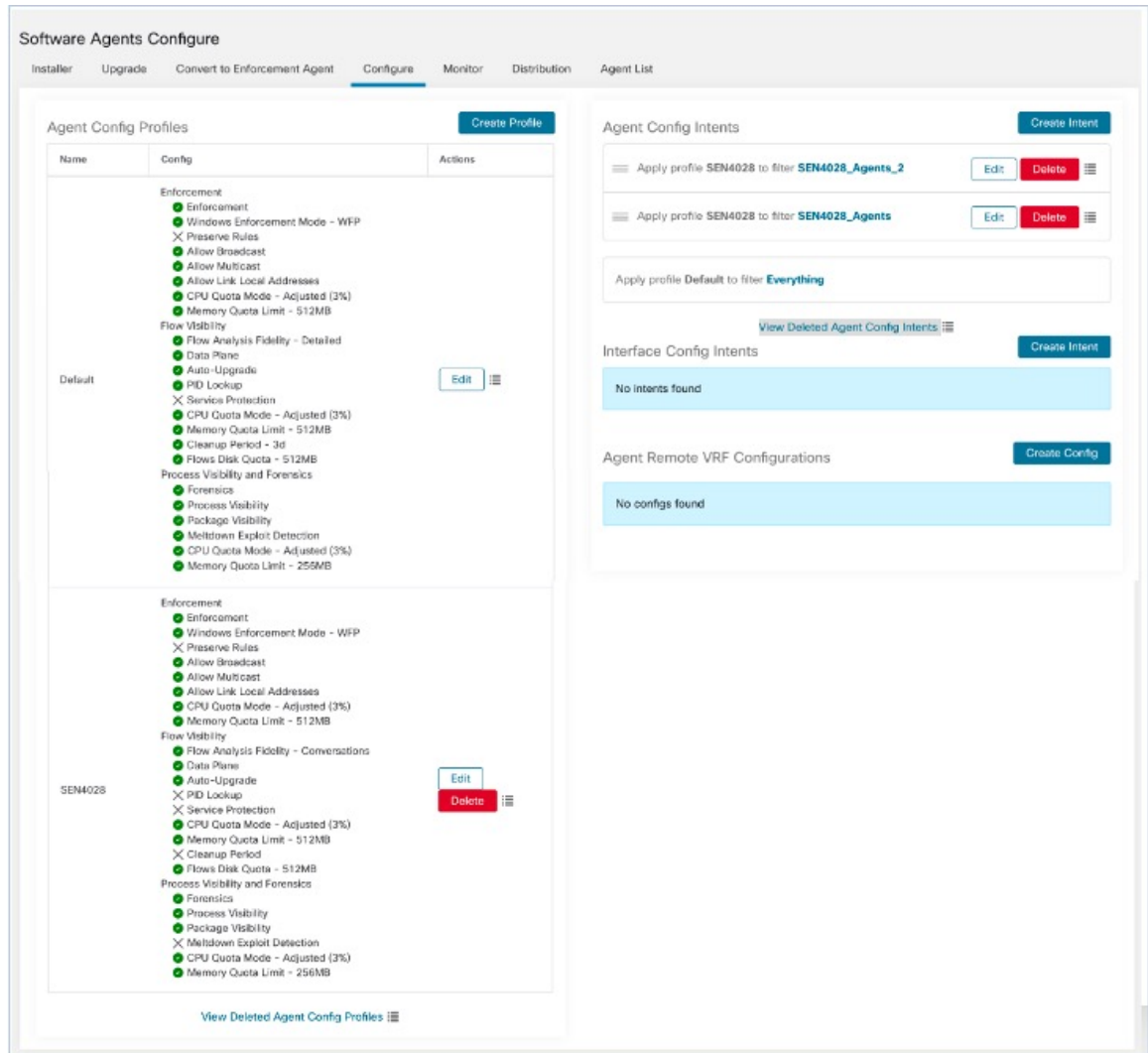
配置软件代理

在“软件代理配置”(Software Agent Configuration)页面上，配置软件代理以创建与资产过滤器或范围关联的意图。对于每个代理，应用第一个匹配的意图。有关详细信息，请参阅[管理 Cisco Secure Workload 的资产](#)。



Note 对于任何 Cisco Secure Workload 部署，请在未与任何特定配置文件关联的所有代理上使用默认代理配置。

Figure 5: 软件代理配置



创建代理配置文件

Before you begin

请参阅 [配置软件代理的要求和前提条件](#), on page 56。

Procedure

- 步骤 1 从导航窗格中，依次选择管理 (Manage) > 工作负载 (Workloads) > 代理 (Agents)。
- 步骤 2 点击配置 (Configure) 选项卡。
- 步骤 3 点击创建配置文件 (Create Profile) 按钮。
- 步骤 4 输入配置文件的名称，并选择配置文件可用的范围。

步骤 5 在下表中列出的字段中输入适当的值。

Table 8: 创建软件代理配置文件字段说明

字段	说明
执行	
执行 (Enforcement)	<p>启用 - 在代理上启用策略执行。启用执行后，代理会执行最近收到的策略集。禁用（默认）- 代理不执行策略。</p> <p>Note 如果在代理上启用、禁用和重新启用策略执行，它会清除防火墙状态并将捕获全部默认操作设置为 ALLOW。</p>
Windows 执行模式 (Windows Enforcement Mode)	<p>在 Windows 工作负载上，代理可以通过以下方式执行网络策略：</p> <ul style="list-style-type: none"> • WFP - Windows 过滤平台（在 Windows 过滤器引擎中直接编写 WFP 过滤器）。 请参阅WFP 模式下 Windows 平台上的代理执行, on page 36。 • WAF（默认）- Windows 高级防火墙。 请参阅WAF 模式下 Windows 平台上的代理执行, on page 34。
保留规则 (Preserve Rules)	<p>启用 - 保留代理上的现有防火墙规则。</p> <p>禁用（默认）- 在应用 Cisco Secure Workload 中的执行策略规则之前清除现有防火墙规则。</p> <p>保存规则属性的行为取决于具体平台。您可以在每个平台的“保留规则” (Preserve Rules) 部分查看属性的详细信息。</p>
允许广播 (Allow Broadcast)	<p>启用（默认）- 向防火墙添加规则，以允许工作负载上的入口和出口广播流量。</p> <p>禁用 - 不添加任何规则。如果代理的默认策略为 DENY，则广播流量就会下降。</p>
允许组播 (Allow Multicast)	<p>启用（默认）- 在防火墙上添加规则，允许工作负载上的多播流量入口和出口。</p> <p>禁用 - 不添加任何规则。如果代理上的默认策略为 DENY，则组播流量会丢弃。</p>
允许链接本地地址 (Allow Link Local Addresses)	<p>启用（默认）- 向防火墙添加规则，以允许工作负载上的链接本地地址流量。</p> <p>禁用 - 不添加任何规则。如果代理上的默认策略为 DENY，则组播流量会丢弃。</p>

字段	说明
CPU 配额模式 (CPU Quota Mode)	<p>已调整（默认）- CPU 限制根据系统中的 CPU 数量进行调整。例如，如果有 10 个 CPU，将 CPU 限制设为 3%，则代理总共只能使用 30%（按顶部计算）。</p> <p>顶部 - CPU 限制值平均与顶部视图匹配。例如，如果将 CPU 限制设置为 3%，而系统中有 10 个 CPU，则 CPU 使用率为 3%。这是一种限制性较强的模式，只有在必要时才使用。</p> <p>禁用 - 禁用 CPU 限制功能。代理会使用操作系统中使用的 CPU 资源。</p> <p>有关详细信息，请参阅 Cisco Secure Workload 产品手册。</p>
CPU 配额限制 (%) (CPU Quota Limit [%])	以系统处理能力的百分比来指定实际限制。
内存配额限制 (MB) (Memory Quota Limit [MB])	指定进程的内存限制 (MB)。如果进程达到此限制，就会重启。
流可视性	
流分析精确度 (Flow Analysis Fidelity)	<p>对话（默认）- 在所有代理上启用对话模式。</p> <p>详细 - 在所有代理上启用详细模式。</p>
数据平面 (Data Plane)	<p>启用（默认）- 启用代理以向集群发送报告。</p> <p>禁用 - 禁用代理的报告。</p>
自动升级 (Auto-Upgrade)	<p>启用（默认）- 有新软件包可用时自动升级代理。</p> <p>禁用 - 不自动升级代理。</p>
PID 查找 (PID Lookup)	<p>启用 - 在代理上启用进程 ID 查找。启用后，代理会尽力尝试将网络流与工作负载中的运行进程相关联。此操作的开销很高，因此代理会控制每个导出周期的操作次数，以控制 CPU 开销。即使启用了配置，某些流也可能未与任何进程关联。</p> <p>禁用（默认）- 不在代理上启用进程 ID 查找。</p>

字段	说明
服务保护 (Service Protection)	<p>启用 - 在代理上启用服务保护。启用后，代理可确保防止用户禁用服务、卸载代理和重启服务。但在禁用服务保护后，您可以继续停止或卸载代理。</p> <p>Note</p> <ul style="list-style-type: none"> 在正常自动升级代理时，请勿禁用服务保护。 不要为代理的手动升级启用服务保护。 服务保护会阻止任何强制升级，如使用安装程序脚本 - forceUpgrade 选项。 在启用服务保护时，任何系统启动的升级均有效。 <p>禁用 (*) - 默认情况下，禁用代理上的服务保护。</p> <p>详细 (默认) - 在所有代理上启用详细模式。</p> <p>Note 此功能仅适用于 Windows 代理。</p>
CPU 配额模式 (CPU Quota Mode)	<p>已调整 (默认) - 根据系统中的 CPU 数量调整 CPU 限制。例如，如果系统中有 10 个 CPU，请将 CPU 限制设置为 3%。</p> <p>选择此模式可允许代理使用总计 30% (按顶部测量)。</p> <p>顶部 - CPU 限制值平均与顶部视图匹配。例如，将系统中 10 个 CPU 的 CPU 限制设置为 3%，则 CPU 使用率仅为 3%。这是一种限制性较强的模式，并且只有在必要时才使用。</p> <p>禁用 - 禁用 CPU 限制功能。代理会使用操作系统中使用的 CPU 资源。</p>
CPU 配额限制 (%) (CPU Quota Limit [%])	指定代理可使用的系统处理能力的实际限制百分比。
内存配额限制 (MB) (Memory Quota Limit [MB])	指定进程允许使用的内存限制 (以 MB 为单位)。如果进程达到此限制，进程将重启。
清理期 (天) (Cleanup period [days])	<p>启用 - 在代理上启用自动清理。输入删除非活动代理之前的天数。</p> <p>禁用 (默认) - 不在代理上启用自动清理。</p>

字段	说明
流磁盘配额 (MB) (Flows Disk Quota [MB])	<p>输入用于存储流数据的最大大小限制 (MB)。</p> <p>如果“流磁盘配额” (Flows Disk Quota) 字段为：</p> <ul style="list-style-type: none"> • 0: 代理不在本地存储离线流。 • 空白: 启用“流时间窗口” (Flows Time Window) 字段。在流时间窗口中输入持续时间后，“流磁盘配额” (Flows Disk Quota) 字段会自动将该值设置为 16 GB。 <p>您可以选择“流磁盘配额” (Flows Disk Quota) 或“流时间窗口” (Flows Time Window) 选项进行流日志缓冲，以防代理和集群之间的连接中断。</p> <p>例如，如果您已将“流时间窗口” (Flows Time Window) 设置为一小时，并且代理无法与集群通信，则代理会存储最后一小时的流数据。本地存储在工作负载上超过一小时的任何流数据都会被更新的日志覆盖。</p> <p>以 MB 为单位指定已存储流数据的总大小限制。</p>
流时间窗口 (小时) (Flows Time Window [Hours])	<p>以小时为单位指定代理必须在本地捕获和存储流的时间。</p> <p>选择流磁盘配额 (Flows Disk Quota) 或流时间窗口 (Flows Time Window)；基于大小或基于时间的轮换。选择“流时间窗口” (Flows Time Window) 后，将“流磁盘配额” (Flows Disk Quota) 设置为 16 GB。将“流磁盘配额” (Flows Disk Quota) 设置为 0 会禁用此功能。</p> <p>当流数据达到大小限制或时间限制时，流数据就会被轮换。</p> <p>仅当未在“流磁盘配额” (Flows Disk Quota) 字段中输入值时，才会显示此字段。</p> <p>以小时为单位输入代理捕获流并将其存储在本地的持续时间。</p> <ul style="list-style-type: none"> • 与代理的连接恢复后，代理会发送实时流数据。 • 在发送实时流数据的同时，代理还会主动上传缓冲遥测数据。遥测数据会以小数据包的形式定期发送。 • 根据缓冲遥测数据的大小和传输速度，发送所有缓冲数据需要多个间隔。 • 代理会逐步删除本地存储的流数据。 <p>在本地存储的过期流数据达到配置的大小或时间限制后，将其删除。</p>
软件包可视性、进程可视性和取证	
软件包可视性	<p>启用 - 启用收集和报告在工作负载中找到的已安装软件包。这是发现和呈现该工作负载所附漏洞 (CVE) 所必需的。</p> <p>禁用 (默认) - 在代理上禁用软件包可视性。</p>

字段	说明
进程可视性	<p>启用 - 启用对长时间运行的进程的跟踪。这是发现和显示工作负载中长期存在的进程列表及其 CPU 和内存消耗趋势以及进程文件散列所必需的。此外，还可构建进程快照图，并将进程与软件包链接起来，从而在进程快照图上显示进程是否与 CVE 相关联。</p> <p>禁用（默认） - 禁用进程可视性。</p>
取证	<p>启用 - 在代理上启用取证。代理将从操作系统中侦听并获取丰富的实时事件。这是检测是否检测到任何已编程取证规则事件所必需的。这也使得代理能够检测和报告短期进程，进而生成更丰富、更全面的进程快照图，并能够在更多捕获的网络流中填充进程命令行。</p>
崩溃漏洞攻击检测	<p>启用 - 在代理上启用取证和崩溃漏洞检测。有关详细信息，请参阅兼容性中的“侧信道”。</p> <p>禁用（默认） - 在代理上禁用崩溃漏洞检测。</p>
CPU 配额模式	<p>已调整（默认） - 根据系统中的 CPU 数量调整 CPU 限制。例如，如果系统中有 10 个 CPU，请将 CPU 限制设置为 3%。选择此模式可使用总计 30%（按顶部测量）。</p> <p>顶部 - CPU 限制值平均与顶部视图匹配。例如，将 CPU 限制设置为 3%，而系统中有 10 个 CPU，则 CPU 使用率保持为 3%。只有在必要时才使用这种限制模式。</p> <p>禁用 - 禁用 CPU 限制功能，代理会使用操作系统允许的 CPU 资源。</p>
CPU 配额限制 (%)	指定代理可使用的系统处理能力的实际限制百分比。
内存配额限制 (MB)	指定内存限制 (MB)。如果存储限制超出指定限制，则进程将重启。

步骤 6 点击保存 (Save)

What to do next

将创建的配置文件与代理配置意图相关联。有关详细信息，请参阅 [创建代理配置意图](#), on page 63。

创建代理配置意图

Before you begin

- 请参阅[配置软件代理的要求和前提条件](#), on page 56。
- 创建代理配置文件。请参阅[创建代理配置文件](#), on page 58。

Procedure

步骤 1 在左侧的导航栏中，点击管理 (Manage) > 代理 (Agents)。

步骤 2 点击配置选项卡。

步骤 3 点击接口配置意图 (Interface Config Intent) 标题旁边的代理配置意图 (Agent Config Intent) 按钮。

步骤 4 在下表中列出的字段中输入适当的值：

字段	说明
配置文件 (Profile) (必填)	输入现有配置文件的名称，然后从下拉菜单中选择该配置文件。
过滤器 (Filter) (必填)	输入现有过滤器或范围的名称，或从下拉菜单中选择创建新过滤器 (Create new filter)。 有关创建过滤器的详细信息，请参阅 过滤器 。

步骤 5 点击保存 (Save)。

Figure 6: 代理配置意图

Agent Config Intents

Apply profile to filter

Apply profile **Default** to filter **Everything**

为代理创建远程 VRF 配置

这是为 Cisco Secure Workload 软件代理分配 VRF 的推荐方法。使用此配置，Cisco Secure Workload 设备会根据与 Cisco Secure Workload 设备的连接上的代理看到的源 IP 地址和源端口将 VRF 分配给软件传感器。

Procedure

步骤 1 在左侧的导航栏中，点击管理 (Manage) > 代理 (Agents)。

步骤 2 点击配置 (Configure) 选项卡。

步骤 3 点击代理远程 VRF 配置 (Agent Remote VRF Configurations) 标题旁边的创建配置 (Create Config) 按钮。

步骤 4 在字段中输入相应的值，然后点击保存 (Save)。

Figure 7: 远程 VRF 配置

Agent Remote VRF Configurations

创建接口配置意图

建议在使用远程 VRF 配置设置中为代理分配虚拟路由和转发 (VRF)。在极少数情况下，当代理主机有多个必须分配给不同 VRF 的接口时，可以选择使用接口配置意图为它们分配 VRF。

Procedure

步骤 1 导航至管理 (Manage) > 代理 (Agents)。

步骤 2 点击配置 (Configure) 选项卡。

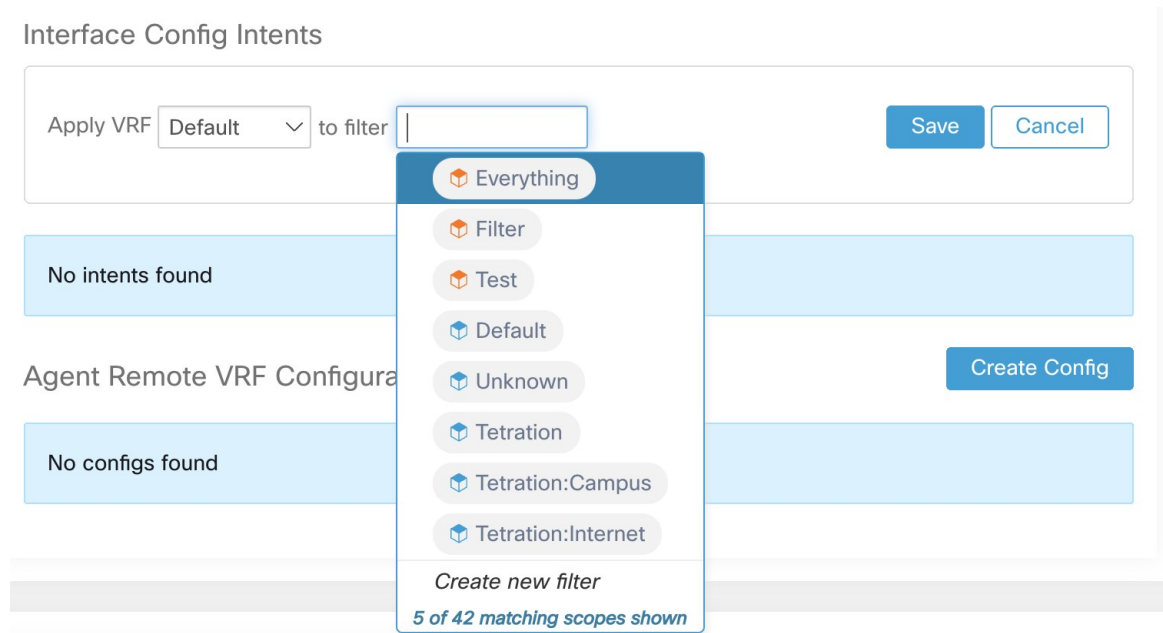
步骤 3 点击接口配置意图 (Interface Config Intent) 标题旁边的创建意图 (Create Intent) 按钮。

步骤 4 在表中列出的字段中输入适当的值：

字段	说明
VRF	从下拉列表中选择 VRF (必填)。
Filter	输入现有过滤器或范围的名称，或从下拉列表中选择创建新过滤器 (Create a new filter) (必填)。有关详细信息，请参阅 过滤器 。

步骤 5 点击保存 (Save)。

Figure 8: 接口配置意图



Note 当您删除具有更高优先级配置意图的接口时，代理不会回退到默认“全部捕获”意图。

在工作负载配置文件中查看详细的代理状态

Procedure

- 步骤 1 请按照上述步骤检查代理状态。
- 步骤 2 在“执行代理” (Enforcement Agents) 页面上，点击代理操作系统分发 (Agent OS Distribution)。选择操作系统，然后点击框右上角的过滤器映像。
- 步骤 3 “软件代理列表” (Software Agent List) 页面上将列出具有所选操作系统分布的代理。
- 步骤 4 点击代理 (Agent) 以查看代理详细信息，然后点击 IP 地址。在“工作负载配置文件” (Workload Profile) 页面上，您可以查看主机配置文件、代理配置文件和代理特定详细信息的详细信息，例如带宽、长期存在的进程、软件包、进程快照、配置、接口、统计信息、策略、容器策略等。
- 步骤 5 点击配置 (Config) 选项卡以查看终端主机上的配置。
- 步骤 6 点击策略 (Policies) 选项卡以查看终端主机上已执行的策略。

Figure 9: 工作负载配置文件 - 配置

Figure 10: 工作负载配置文件 - 策略

Priority	Packets	Bytes	Actions	Direction	Family	Proto	Src Inventory	Src Ports	Dest Inventory	Dest Ports
1	N/A	N/A	ALLOW	INGRESS	IPv4	TCP	any	any	172.21.95.163/32	22
2	N/A	N/A	ALLOW	EGRESS	IPv4	TCP	172.21.95.163/32	22	any	any
3	N/A	N/A	ALLOW	INGRESS	IPv4	TCP	any	22	172.21.95.163/32	any
4	N/A	N/A	ALLOW	EGRESS	IPv4	TCP	172.21.95.163/32	any	any	22
5	N/A	N/A	ALLOW	INGRESS	IPv4	ST	ubuntuhosts	any	172.21.95.163/32	any
6	N/A	N/A	ALLOW	EGRESS	IPv4	ST	172.21.95.163/32	any	ubuntuhosts	any
7	N/A	N/A	ALLOW	INGRESS	IPv4	ST	ubuntuhosts	any	172.21.95.163/32	any
8	N/A	N/A	ALLOW	EGRESS	IPv4	ST	172.21.95.163/32	any	ubuntuhosts	any
9	N/A	N/A	ALLOW	INGRESS	IPv4	STP	ubuntuhosts	any	172.21.95.163/32	any
10	N/A	N/A	ALLOW	EGRESS	IPv4	STP	172.21.95.163/32	any	ubuntuhosts	any
11	N/A	N/A	ALLOW	INGRESS	IPv4	STP	ubuntuhosts	any	172.21.95.163/32	any
12	N/A	N/A	ALLOW	EGRESS	IPv4	STP	172.21.95.163/32	any	ubuntuhosts	any
13	N/A	N/A	ALLOW	INGRESS	IPv4	SUNND	ubuntuhosts	any	172.21.95.163/32	any

Note Windows 代理主机不支持获取所有统计信息 (Fetch All Stats)，后者用于为单个策略提供统计信息。

重新连接代理

代理重新连接是将用户从本地部署转移到 SaaS 或 SaaS 转移到本地部署的方法。

用户角色

- 站点管理员
- 客户支持代表

您可以迁移到 SaaS 环境，也可以从 SaaS 环境迁移，尤其是从 SaaS 环境迁移到本地部署环境时，您必须与内部支持团队合作。

工作流程

- 输入激活密钥、传感器虚拟 IP 和传感器证书颁发机构 (CA) 和 [启用重新连接](#), on page 68。
- [选择要重新连接的代理](#), on page 70。
- [禁用重新连接](#), on page 70。



Note 在任何时候，您都只能将代理移动到一个目标。建议您在移动代理后禁用代理重新连接。

启用重新连接

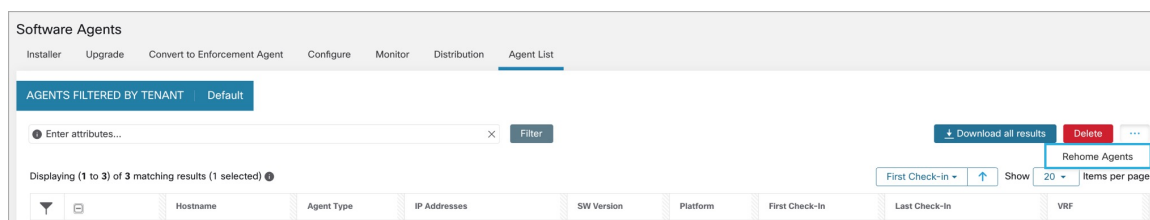
Procedure

步骤 1 在左侧导航菜单中，点击**管理 (Manage) > 工作负载 (Workloads) > 代理 (Agents)**。

步骤 2 点击代理列表 (**Agent List**) 选项卡。

步骤 3 点击菜单图标，然后选择**重新连接代理 (Rehome Agents)**。

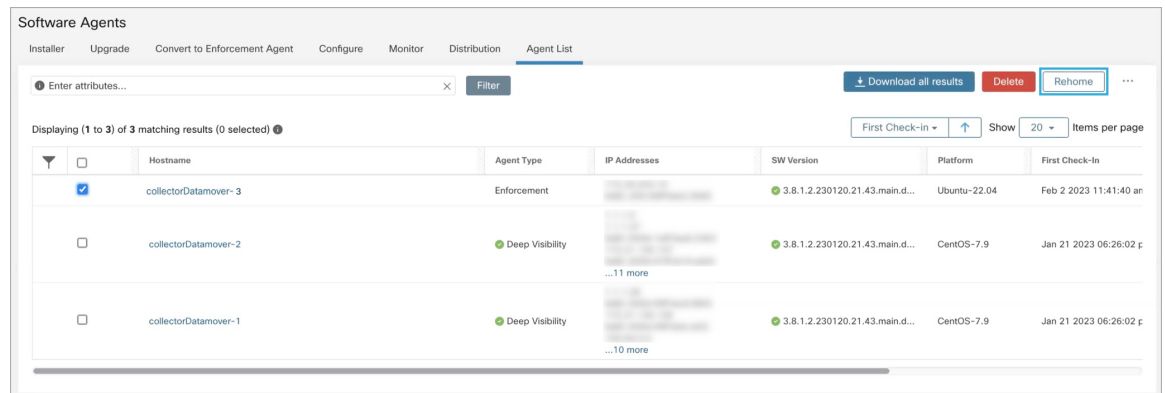
Figure 11: 重新连接代理



步骤 4 在代理重新连接 (**Agent Rehomeing**) 窗口中，填写以下详细信息：

字段	说明
目标范围激活密钥	<ol style="list-style-type: none"> 导航至管理 (Manage) > 工作负载 (Workloads) > 代理 (Agents)。 点击安装程序 (Installer) 选项卡。 选择使用经典打包安装程序手动安装 (Manual install using classic packaged installers)。 点击下一步 (Next)。 点击代理激活密钥 (Agent Activation Key)。 复制密钥值并将其粘贴到目的范围激活密钥 (Destination Scope Activation Key) 字段中。
目标传感器 VIP	<ol style="list-style-type: none"> 导航至平台 (Platforms) > 集群配置 (Cluster Configuration)。 复制传感器 VIP 并将其粘贴到目标传感器 VIP (Destination Sensor VIP) 字段中。
HTTPS 代理	如果代理需要，请输入代理域或地址以使用代理进行出站通信。
目标传感器 CA 证书	<ol style="list-style-type: none"> 导航至平台 (Platforms) > 集群配置 (Cluster Configuration)。 点击下载传感器 CA 证书 (Download Sensor CA Cert)。

Figure 12: 启用代理重新连接



步骤 5 点击启用代理重新连接 (Enable Agent Rehomng)。

配置已保存。“重新连接”(Rehome)按钮显示在右上角。

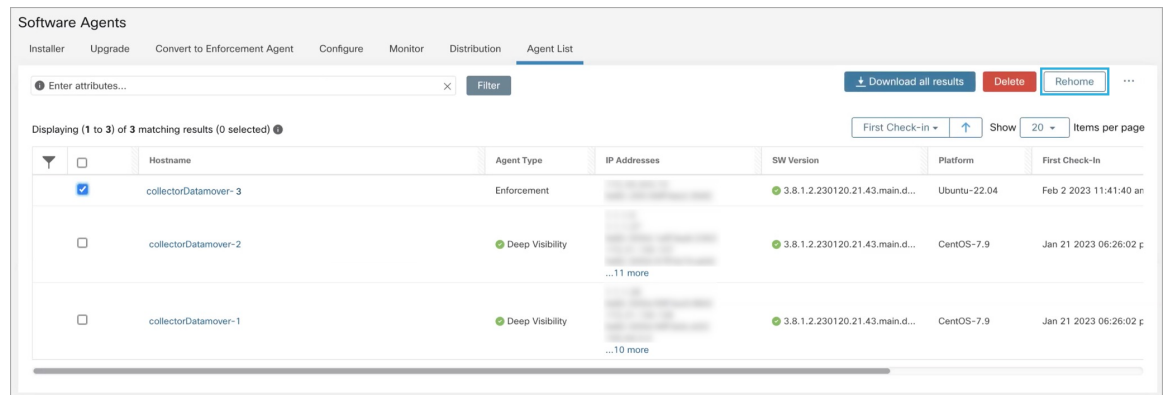
选择要重新连接的代理

Procedure

步骤 1 选择一个代理。

步骤 2 点击重新连接 (Rehome)。

Figure 13: 选择要重新连接的代理



步骤 3 点击是进行确认。

禁用重新连接



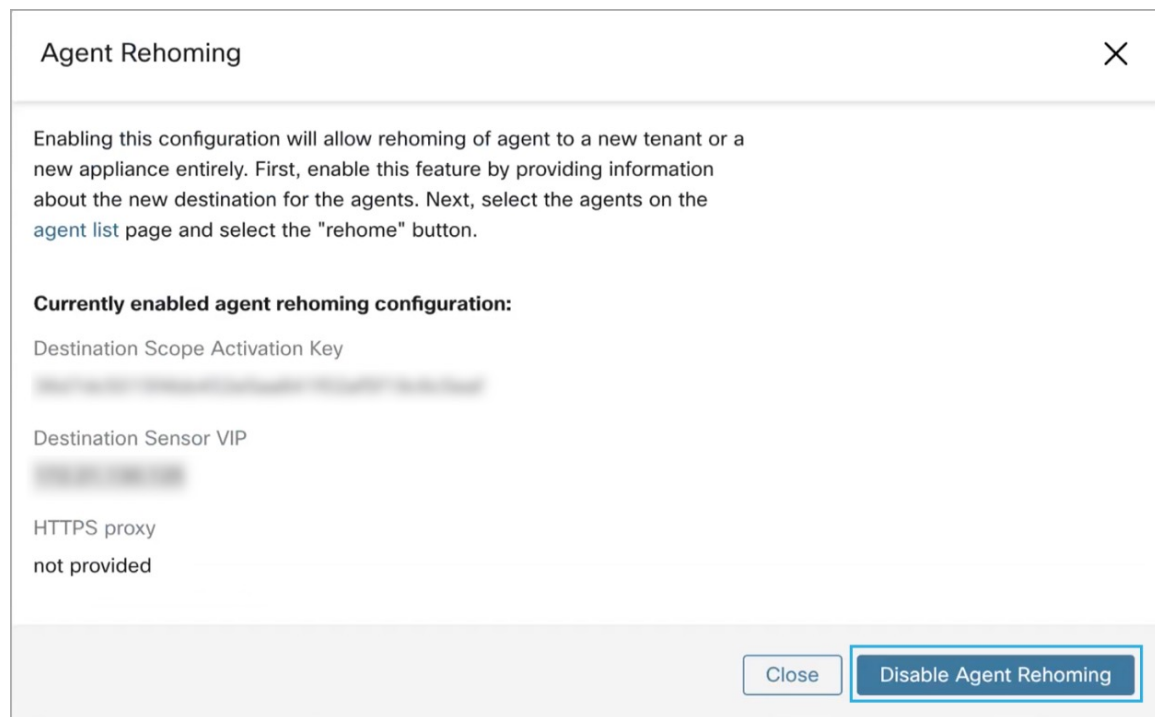
Note 如果有多个用户重新连接到 SaaS 或从 SaaS 重新连接，则站点管理员必须分别移动每个租户或设备。要执行此操作，请禁用重新连接以清除设置，然后为新用户启用重新连接。

Procedure

步骤 1 点击菜单图标并选择重新连接代理 (Rehome Agents)。

步骤 2 在“代理重新连接”(Agent Rehoming)窗口中，点击禁用代理重新连接(Disable Agent Rehoming)。

Figure 14: 禁用代理重新连接



生成代理令牌

在代理配置文件中，您可以启用服务保护，以防止卸载、禁用和停止 Windows 代理服务。要对代理执行任何更改，可以在代理配置文件上禁用此保护。但是，如果由于连接问题而无法禁用保护，则可以生成代理令牌以禁用工作负载上的服务保护。令牌有效期为 15 分钟。

支持生成和检索代理令牌的角色：

- **站点管理员：**适用于集群或租户。
- **客户支持：**适用于租户。
- **代理安装程序：**用于代理特定的令牌。



Note 您只能为基于 Windows 操作系统的软件代理生成基于时间的代理令牌。

要生成和下载代理令牌，请执行以下步骤：

Procedure

步骤 1 在导航窗格中，点击**管理 (Manage)**>**工作负载 (Workloads)**>**代理 (Agents)**>**代理列表 (Agent List)**。
根据要求，您可以选择代理令牌类型之一 - 集群、租户或特定代理。对于代理特定令牌，请转至步骤 5。

步骤 2 点击菜单图标并选择**代理令牌 (Agent Token)**。

Note **代理令牌 (Agent Token)** 选项仅对站点管理员或客户支持用户角色可见。

步骤 3 选择令牌类型：

- 集群令牌 (Token For Cluster) - 此选项仅对站点管理员可见，并且令牌适用于所有代理。
- 租户令牌 (Token For Tenant) - 适用于所选租户下的代理。

步骤 4 要下载令牌密钥，请点击**下载令牌 (Download Token)**。

步骤 5 要查看和下载特定代理的令牌密钥详细信息，请执行以下操作：

- 转到**代理列表 (Agent List)** 选项卡，然后点击所需的代理。在**代理详细信息 (Agent Details)**>**代理令牌 (Agent Token)** 下，您可以查看令牌的令牌密钥和到期详细信息。
- 要下载代理特定的令牌，请点击**下载令牌 (Download Token)**。

What to do next

下载代理令牌文件后，在代理上运行以下命令以禁用服务保护：`"C:\Program Files\Cisco Tetration\TetSen.exe" -unprotect <token>`，其中 `token` 是下载的代理令牌。

使用令牌禁用服务保护后，服务保护可能会在服务重启并连接到 Cisco Secure Workload 集群时自动重新启用。

启用执行时主机 IP 地址更改

作为站点管理员，如果在代理上已启用执行的情况下更改主机的 IP 地址，则在主机防火墙规则中发现该主机 IP 地址且全部捕获设置为“拒绝”时，可能会产生影响。

在这种情况下，请执行以下步骤来更改主机 IP 地址：

Procedure

步骤 1 在 Cisco Secure Workload UI 上，[创建新代理配置文件](#)并禁用执行。

步骤 2 创建包含需要更改 IP 地址的主机及其旧地址和新地址的列表。

步骤 3 将新创建的代理配置文件应用于意图并保存意图。

步骤 4 选择要更改其 IP 地址的主机，并确保已禁用这些主机的执行。

- 步骤 5 更改所选主机的 IP 地址。
 - 步骤 6 在 Cisco Secure Workload UI 上，通过包含这些主机的新 IP 地址来更新范围内的过滤器。
 - 步骤 7 在接口 (**Interfaces**) > 代理工作负载配置文件 (**Agent Workload Profile**) 选项卡下，验证 IP 地址是否已更改为新的 IP 地址。
 - 步骤 8 在策略 (**Policies**) 选项卡下，确保使用新 IP 地址生成策略。
 - 步骤 9 删除之前创建的意图或配置文件。
 - 步骤 10 点击启用执行 (**Enable Enforcement**)，在已禁用执行的较早代理配置配置文件的范围内启用执行。
-

升级软件代理

从 UI 升级代理

可以使用代理配置意图工作流程升级代理，如下所述：[软件代理配置](#)。配置代理配置文件时，有一个可以启用或禁用的**自动升级 (Auto Upgrade)** 选项。如果启用该选项，符合库存过滤条件的代理将自动升级到最新可用版本。

在软件代理 (**Software Agents**) > 代理列表 (**Agent List**) 页面上，具有过期版本的软件代理会在**软件版本 (SW Version)** 列下突出显示，并带有警告标志。务必将这些代理升级到集群上的最新可用版本。

要使用软件代理配置意图工作流程来配置软件代理升级，请执行以下操作：

Procedure


- 步骤 1 在资产过滤器 (**Inventory Filters**) 页面上创建资产过滤器。有关详细信息，请参阅[过滤器](#)。

Figure 15: 资产过滤器

+ Create an Inventory Filter

1 Define ————— 2 Summary


Name


Development Linux VMs 

Create a query based on Inventory Attributes:




Inventory is matched dynamically based on the query. The labels can include Hostname, Address/Subnet, OS, and more. The [full list](#) is in the user guide.

A preview of matching inventory items will be shown in the next step.

Query 

Hostname contains linux 

[Show advanced options](#)

步骤 2 为资产过滤器选择的代理创建代理配置文件。或者，您可以启用**自动升级 (Auto Upgrade)** 选项以自动升级所选代理。

Figure 16: 代理配置

Agent Config Profiles Create Profile

Name ↑	Config	Actions
Default	Enforcement <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Enforcement <input checked="" type="checkbox"/> Windows Enforcement Mode - WAF <input type="checkbox"/> Preserve Rules <input checked="" type="checkbox"/> Allow Broadcast <input checked="" type="checkbox"/> Allow Multicast <input checked="" type="checkbox"/> Allow Link Local Addresses <input checked="" type="checkbox"/> CPU Quota Mode - Adjusted (3%) <input checked="" type="checkbox"/> Memory Quota Limit - 512MB Flow Visibility <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Flow Analysis Fidelity - Detailed <input checked="" type="checkbox"/> Data Plane <input checked="" type="checkbox"/> Auto-Upgrade <input type="checkbox"/> PID Lookup <input checked="" type="checkbox"/> CPU Quota Mode - Adjusted (3%) <input checked="" type="checkbox"/> Memory Quota Limit - 512MB Process Visibility and Forensics <ul style="list-style-type: none"> <input type="checkbox"/> Forensics <input type="checkbox"/> Meltdown Exploit Detection <input checked="" type="checkbox"/> CPU Quota Mode - Adjusted (3%) <input checked="" type="checkbox"/> Memory Quota Limit - 256MB 	Edit
VM	Enforcement <ul style="list-style-type: none"> <input type="checkbox"/> Enforcement <input checked="" type="checkbox"/> Windows Enforcement Mode - WAF <input type="checkbox"/> Preserve Rules <input checked="" type="checkbox"/> Allow Broadcast <input checked="" type="checkbox"/> Allow Multicast <input checked="" type="checkbox"/> Allow Link Local Addresses <input checked="" type="checkbox"/> CPU Quota Mode - Adjusted (3%) <input checked="" type="checkbox"/> Memory Quota Limit - 512MB Flow Visibility <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Flow Analysis Fidelity - Detailed <input checked="" type="checkbox"/> Data Plane <input checked="" type="checkbox"/> Auto-Upgrade <input type="checkbox"/> PID Lookup <input checked="" type="checkbox"/> CPU Quota Mode - Adjusted (3%) <input checked="" type="checkbox"/> Memory Quota Limit - 512MB Process Visibility and Forensics <ul style="list-style-type: none"> <input type="checkbox"/> Forensics <input type="checkbox"/> Meltdown Exploit Detection <input checked="" type="checkbox"/> CPU Quota Mode - Adjusted (3%) <input checked="" type="checkbox"/> Memory Quota Limit - 256MB 	Edit Delete

[View Deleted Agent Config Profiles](#)

步骤 3 创建代理配置意图，以便将配置文件应用于使用资产过滤器选择的代理。如果启用自动升级选项，则所选代理会自动升级。

应用代理配置文件后，升级代理通常最多需要 30 分钟。

Figure 17:代理配置意图

Agent Config Intents

Apply profile to filter

Apply profile **Default** to filter **Everything**

Note 默认代理配置文件中的自动升级设置适用于 ERSPAN。

手动代理升级

以下部分将介绍如何在不使用传感器配置意图 workflow 的情况下手动升级代理。

Procedure

步骤 1 在左侧导航窗格中，点击**管理 (Manage) > 工作负载 (Workloads) > 代理 (Agents)**。

步骤 2 点击**升级 (Upgrade)** 选项卡。

系统将显示深度可视性和执行代理，每个代理只列出可升级到的较新版本。默认情况下，系统会选择最新版本。

步骤 3 要过滤特定代理，请在过滤框中输入搜索查询。例如，输入 Platform = CentOS-7.6。

步骤 4 选择要升级到所选版本的代理，然后点击**升级 (Upgrade)**。

Note 在正常情况下，强烈建议允许代理自动升级，这是唯一支持的升级方法。如果想通过手动下载最新版本并直接部署到工作负载上运行的代理来控制升级，请确保遵循安全预防措施。

Kubernetes/OpenShift 代理的升级行为

使用守护进程集安装程序脚本安装在 Kubernetes/OpenShift 节点上的代理能够自行升级。升级过程由自动升级选项或手动触发 Kubernetes/OpenShift 集群中任何节点的升级来控制。此环境下的升级机制是升级守护进程集规范中的 Docker 映像，这意味着一个代理的升级会影响守护进程集涵盖的所有代理，下一段将对此进行解释。

当守护程序集 Pod 规格发生更改时，Kubernetes/OpenShift 将触发正常关闭，获取新的 Docker 映像并启动 Kubernetes/OpenShift 集群中所有节点上的 Cisco Secure Workload 代理 Pod。这将导致即使允许升级的策略只适用于集群中的一个节点子集，其他节点上的代理也会被升级。

如果禁用了所有节点的自动升级，则可以通过下载新的安装程序脚本并重新运行安装来进行手动升级。安装脚本会自动检测新安装与升级现有安装的情况，并在检测到已有安装时手动升守护程序集 Pod。

删除软件代理

删除深度可视性或执行 Linux 代理

基于 RPM 的安装：

1. 运行命令：`rpm -e tet-sensor`

代理卸载事件会传达给集群，并且代理将在软件代理 (Software Agent) 页面上标记为已卸载。

在软件代理 (Software Agent) 页面上从 UI 手动删除代理，或者用户可以通过在代理配置文件中启用清理期 (cleanup period) 来启用代理的自动清理或删除。



Note 默认情况下，清理期 (cleanup period) 处于关闭状态。

基于 Ubuntu .deb 的安装：

全新安装的 Ubuntu 代理现在将使用本地 .deb 格式。

1. 运行命令：`dpkg -purge tet-sensor`

代理卸载事件会传达给集群，并且代理将在软件代理 (Software Agent) 页面上标记为已卸载。

在软件代理 (Software Agent) 页面上从 UI 手动删除代理，或者用户可以通过在代理配置文件中启用清理期 (cleanup period) 来启用代理的自动清理或删除。



-
- Note**
- 默认情况下，清理期 (cleanup period) 处于关闭状态。
 - 在代理操作期间，内核可能会自动加载某些内核模块。例如，如果在 Linux 中启用了执行，则可能会加载 Netfilter 模块。代理没有内核加载的模块列表。因此，在代理卸载期间，它不可能卸载内核模块。
 - 如果执行代理将策略应用于系统防火墙，则卸载代理会清除应用的策略并打开系统防火墙。
-

Figure 18: 代理卸载警报

Hostname	Agent Type	IP Addresses	SW Version	Platform	First Check-in	Last Check-in	WRF
b4-ai-hy-centos76	Enforcement	172.28.231.175 fe80:250:56f:fe91:1b2b5 172.20.207.106 fe80:a5fa:c5ac:b5e5:e097 192.168.122.1	3.8.1.2.2301.3021-enforcer	OracleSolaris-11.4	Feb 9 2023 02:59:20 pm (PST)	Feb 9 2023 08:59:44 pm (PST)	Default
sensor-dev-rocky90	Enforcement	10.195.210.122 fe80:250:56f:fe91:ca35	3.8.1.2.230130.21.43.main.dev-e...	RockyLinux-9.0	Feb 3 2023 12:02:31 am (PST)	Feb 9 2023 09:01:48 pm (PST)	Default
sensor-dev-oracle9	Enforcement	10.195.210.121 fe80:250:56f:fe91:1c2d	3.8.1.2.230130.21.43.main.dev-e...	OracleServer-9.0	Feb 3 2023 12:01:09 am (PST)	Feb 9 2023 09:23:27 pm (PST)	Default
sensor-dev-almalinux9	Enforcement	10.195.210.120 fe80:250:56f:fe91:53b8	3.8.1.2.230130.21.43.main.dev-e...	AlmaLinux-9.0	Feb 3 2023 12:00:00 am (PST)	Feb 9 2023 09:22:52 pm (PST)	Default
hartmut-u16	Enforcement	172.26.231.235 fe80:250:56f:fe91:34c4	3.7.1.5.dev-el-enforcer	Ubuntu-16.04	Feb 2 2023 11:27:44 am (PST)	Feb 9 2023 09:19:46 pm (PST)	Default
p91-insu06	Enforcement	172.28.157.105 fe80:288e:97fe:fe3e:5902	3.8.1.2.230130.21.43.main.dev-e...	AlmaLinux-9.0	Feb 2 2023 08:46:08 am (PST)	Feb 9 2023 09:20:33 pm (PST)	Default
sensor-dev-deb11	Enforcement	172.20.207.223 fe80:250:56f:fe91:a737	3.8.1.2.230130.21.43.main.dev-e...	Debian-11	Feb 2 2023 08:44:43 am (PST)	Feb 9 2023 09:21:10 pm (PST)	Default
agent-reg-deb10	Enforcement	10.195.210.123 fe80:250:56f:fe91:13d6	3.8.1.2.230130.21.43.main.dev-e...	Debian-10	Feb 2 2023 08:43:18 am (PST)	Feb 9 2023 09:18:56 pm (PST)	Default
sensor-dev-deb9	Enforcement	10.195.210.189 fe80:250:56f:fe91:c542	3.8.1.2.230130.21.43.main.dev-e...	Debian-9	Feb 2 2023 08:41:18 am (PST)	Feb 9 2023 09:19:10 pm (PST)	Default
sensor-dev-deb8	Enforcement	10.195.210.145 fe80:250:56f:fe91:d8a4	3.8.1.2.230130.21.43.main.dev-e...	Debian-8	Feb 2 2023 08:39:05 am (PST)	Feb 9 2023 09:21:08 pm (PST)	Default
p91-asa09	Enforcement	172.29.157.24 fe80:288e:96f:fe09:9202 ca614:9d19	3.8.1.2.230130.21.43.main.dev-e...	AIX-7.2	Feb 1 2023 06:44:31 am (PST)	Feb 9 2023 09:08:44 pm (PST)	Default
collectorDatacenter-1	Deep Visibility	1.1.1.26 fe80:5054:aaff:fe20:bd3c 10.195.245.22 fe80:205a:b6f:fe9b:b306 100.64.1.0 15 moon	3.8.1.2.230130.21.43.main.dev-s...	CentOS-7.9	Jan 31 2023 08:08:47 pm (PST)	Feb 9 2023 09:09:39 pm (PST)	Tetration Default

删除深度可视性或执行 Windows 代理

有两个选项可用于卸载 Cisco Secure Workload 代理：

Procedure

步骤 1 导航至控制面板 (Control Panel) > 程序 (Programs) > 程序和功能 (Programs and Features)，然后卸载 Cisco Secure Workload 代理。

步骤 2 或者，在内部运行快捷方式 **Uninstall.lnk**。

```
C:\Program Files\Cisco Tetration
```

步骤 3 如果执行代理将策略应用于系统防火墙，则卸载代理会清除应用的策略，并打开系统防火墙。

卸载代理后，系统会更新集群信息。系统将在软件代理 (Software Agent) 页面上更新代理状态，并将代理标记为已卸载。

在软件代理 (Software Agent) 页面上从 UI 手动删除代理，或者可以通过在代理配置文件中激活清理期 (cleanup period) 来启用代理的自动清理或删除。

Note 默认情况下，清理期会保持关闭。

Note

- 如果您在代理安装期间安装了 Npcap，则在卸载代理时也会卸载 Ncap。
- 默认情况下，卸载过程中不会删除日志文件、配置文件和证书。如果要删除它们，请运行同一文件夹中的快捷方式 **UninstallAll.lnk**。

删除深度可视性或执行 AIX 代理

Procedure

运行命令： `installp -u tet-sensor` 。

代理卸载事件将传达给集群，并且代理将在软件代理 (Software Agent) 页面上标记为已卸载。

在软件代理 (Software Agent) 页面上从 UI 手动删除代理，或者用户可以通过在代理配置文件中启用清理期 (cleanup period) 来启用代理的自动清理或删除。

- Note**
- 默认情况下，清理期 (cleanup period) 处于关闭状态。
 - 深度可视性代理由系统资源控制器作为 tet-sensor 进行控制。可以启动、停止、重启和删除它。该服务由 inittab 作为 tet-sen-engine 持续提供。
 - 执行代理由系统资源控制器作为 tet-enforcer 进行控制。可以启动、停止、重启和删除它。该服务由 inittab 作为 tet-sen-engine 持续提供。
 - 在代理操作期间，内核可能会自动加载某些内核模块。例如，如果在 AIX 中启用了执行，则会加载 ipfilter 模块。代理没有内核加载的模块列表。因此，在代理卸载期间，它不可能卸载内核模块。
 - 如果执行代理将策略应用于系统防火墙，则卸载代理会清除应用的策略并打开系统防火墙。
-

删除通用 Linux 代理

Procedure

步骤 1 运行卸载脚本： `/usr/local/tet-light/uninstall.sh` 。

步骤 2 在软件代理 (Software Agent) 页面上从 UI 删除代理

删除通用 Windows 代理

Procedure

步骤 1 运行卸载脚本： `C:\Program Files\Cisco Tetration\Lightweight Sensor\uninstall.cmd` 。

步骤 2 在软件代理 (Software Agent) 页面上从 UI 删除代理

删除执行 Kubernetes 或 OpenShift 代理

Procedure

步骤 1 找到原始安装程序脚本或从 Cisco Secure Workload UI 下载新的脚本。

步骤 2 运行卸载选项：**install.sh - uninstall**。安装期间的注意事项相同。

- 仅在 Linux x86_64 架构上支持。
- 要么 `~/kube/config` 包含管理员凭证用户，要么使用 `-kubeconfig` 选项指向 `kubectl` 管理员凭证文件。

步骤 3 从软件代理 (Software Agent) 页面上的 UI 删除所有 Kubernetes 节点的代理

删除深度可视性 Solaris 代理

Procedure

步骤 1 运行命令：

- 对于 Solaris 11.4: `pkg uninstall tet-sensor`
- 对于 Solaris 10: `pkgrm -a /opt/cisco/secure-workload/noask.admin -n tet-sensor`

步骤 2 在软件代理 (Software Agent) 页面上删除代理。

工作负载代理收集和导出的数据

本部分介绍软件代理的主要组件、软件代理如何注册到后端服务、收集哪些数据并将其导出到集群以进行分析。

注册

代理成功安装到系统后，需要向后台服务注册，以获得有效的唯一标识符。注册请求中会发送以下信息：

- 主机名
- BIOS-UUID
- 平台信息（例如 CentOS-6.5）
- 自生成客户端证书（使用 `openssl` 命令生成）
- 代理类型（可视性或执行。）

如果代理无法从服务器上获得有效的 ID，它将不断重试，直到获得为止。注册代理非常重要，否则，与其他服务（如采集器）的所有后续通信都将被拒绝。

代理升级

代理会定期（大约 30 分钟）向后端服务发送消息，以报告其当前版本。后台服务使用代理的 ID 及其当前版本来决定该代理是否有新的软件包。发送以下信息：

- 代理的 ID（成功注册后获取）
- 当前代理的版本

配置服务器

代理会将以下信息导出到已配置的配置服务器：

- 主机名
- 代理的 ID（成功注册后获取）
- 接口列表，每个接口包括：
 1. 接口的名称
 2. IP 系列（IPv4 或 IPv6）
 3. IP 地址
 4. 网络掩码
 5. Mac 地址
 6. 接口的索引

一旦任何接口属性发生变化（如现有接口的 IP 地址发生变化，或有新接口出现），该列表就会被刷新并报告给配置服务器。

网络流信息

网络流信息汇总了流经系统的所有数据包。有两种捕获流信息的模式：“详细”和“对话”。默认情况下，**对话**模式用于捕获流信息。捕获的流会被导出到收集器，导出的信息包括：

- 流标识符：唯一标识网络流。它包括一般信息，例如：IP 协议、源和目标 IP 以及第 4 层端口。
- IP 信息：包含在 IP 标头中看到的信息，例如：TTL、IP 标志、数据包 ID、IP 选项和分段标志。
- TCP 信息：包含在 TCP 标头中看到的信息，例如：序列号、确认号、TCP 选项、接收窗口大小。
- 流信息：流统计信息（例如总数据包数、总字节数、TCP 标志统计信息、数据包长度统计信息和套接字统计信息）、从中观察到流的接口索引、流的开始时间和结束时间。
- 在 K8s 环境中，代理可捕获来自 pod 和主机的网络流，然后将这些流关联起来，并报告为相关流。这符合以下 CNI 的条件：
 - Calico
 - Flannel
 - Weave
 - AKS/GKE/AWS VPC CNI
 - Openshift CNI
 - Cilium CNI



Note 从 Pod 和主机捕获网络流，但使用 Cilium CNI 时，无法对网络流进行关联。

在“对话”模式下，代理仅导出本质上是双向的 TCP 流以及其他无连接流。Windows、AIX 和 Linux 平台支持对话模式。有关对话模式的详细信息，请参阅[对话模式](#)。



-
- Note**
- 在 K8s 环境中，Pod 或主机流量的相关性不在“对话”模式下进行。
 - 在任一模式下，代理都不会导出以下流：
 - ARP/RARP 对话
 - 流向收集器的代理流
-

计算机信息

计算机信息描述了主机上运行的所有进程。此外，它还包含与进程相关联的网络信息以及用于启动进程的命令。计算机信息每分钟输出一次，其中包括以下信息：

- 进程 ID
- 用户 ID：流程所有者

- 父进程 ID
- 用于启动进程的命令字符串
- 套接字信息：协议（例如 UDP 或 TCP）、地址类型：IPv4 或 IPv6、源和目标 IP、源和目标端口、TCP 状态、进程的开始和结束时间、进程二进制文件的路径
- 取证信息：有关详细信息，请参阅[兼容性](#)部分。

代理统计信息

代理会跟踪各种统计信息，包括系统和自身的统计信息，例如：

- 代理的开始时间和正常运行时间
- 代理在用户模式和内核模式下的运行时间
- 接收和丢弃的数据包数
- 成功和失败的 SSL 连接数
- 总流数据包数和字节数
- 导出到收集器的流和数据包总数
- 代理的内存和 CPU 使用率

执行警报

执行警报分为三种类型：

- 代理可访问性

此警报会检测代理何时无法访问。如果代理未与 Cisco Secure Workload 集群通信的时间超过配置的秒数，则会触发此警报。

Configure Enforcement Alerts ✕

Configured Alerts

- Scope: **Tetration** when **Agent not reachable (seconds) > 300**
- Scope: **Tetration** when **Firewall = Off**
- Scope: **Tetration** when **Policy = Deviated**

[More details ...](#)

Types

Agent Reachability ⓘ
 Workload Firewall ⓘ
 Workload Policy ⓘ

For Scope: **Tetration**

Agent not reachable (seconds) > 3000 ✕

Severity

Hide Advanced Settings ^

Individual Alerts

Summary Alerts

- 工作负载防火墙

如果在工作负载上配置了执行，但检测到工作负载防火墙已关闭，则会触发此警报，因为此条件将阻止 Cisco Secure Workload 代理执行流量策略。

Configure Enforcement Alerts

Configured Alerts

- Scope: **Tetration** when **Agent not reachable (seconds) > 300**
- Scope: **Tetration** when **Firewall = Off**
- Scope: **Tetration** when **Policy = Deviated**

[More details ...](#)

Types

Agent Reachability ⓘ Workload Firewall ⓘ Workload Policy ⓘ

For Scope: **Tetration**

Firewall is Off ⓘ

Severity

Low Medium High Critical Immediate Action

Hide Advanced Settings ^

Individual Alerts

Enable Disable

Summary Alerts

None Hourly Daily

[Dismiss](#) [Create](#)

- 工作负载策略

如果工作负载防火墙规则不同于适用于此工作负载的 Cisco Secure Workload 策略（工作负载的“具体策略”），则会触发此警报。

Configure Enforcement Alerts

Configured Alerts

- Scope: **Tetration** when **Agent not reachable (seconds) > 300**
- Scope: **Tetration** when **Firewall = Off**
- Scope: **Tetration** when **Policy = Deviated**

[More details ...](#)

Types

Agent Reachability ⓘ Workload Firewall ⓘ Workload Policy ⓘ

For Scope: **Tetration**

Policy is Deviated ⓘ

Severity

Low Medium High Critical Immediate Action

Hide Advanced Settings ^

Individual Alerts

Enable Disable

Summary Alerts

None Hourly Daily

您可以设置警报的严重性以及每个类型的配置参数。

要配置执行警报，请参阅[配置告警](#)。

Figure 19: 配置执行警报

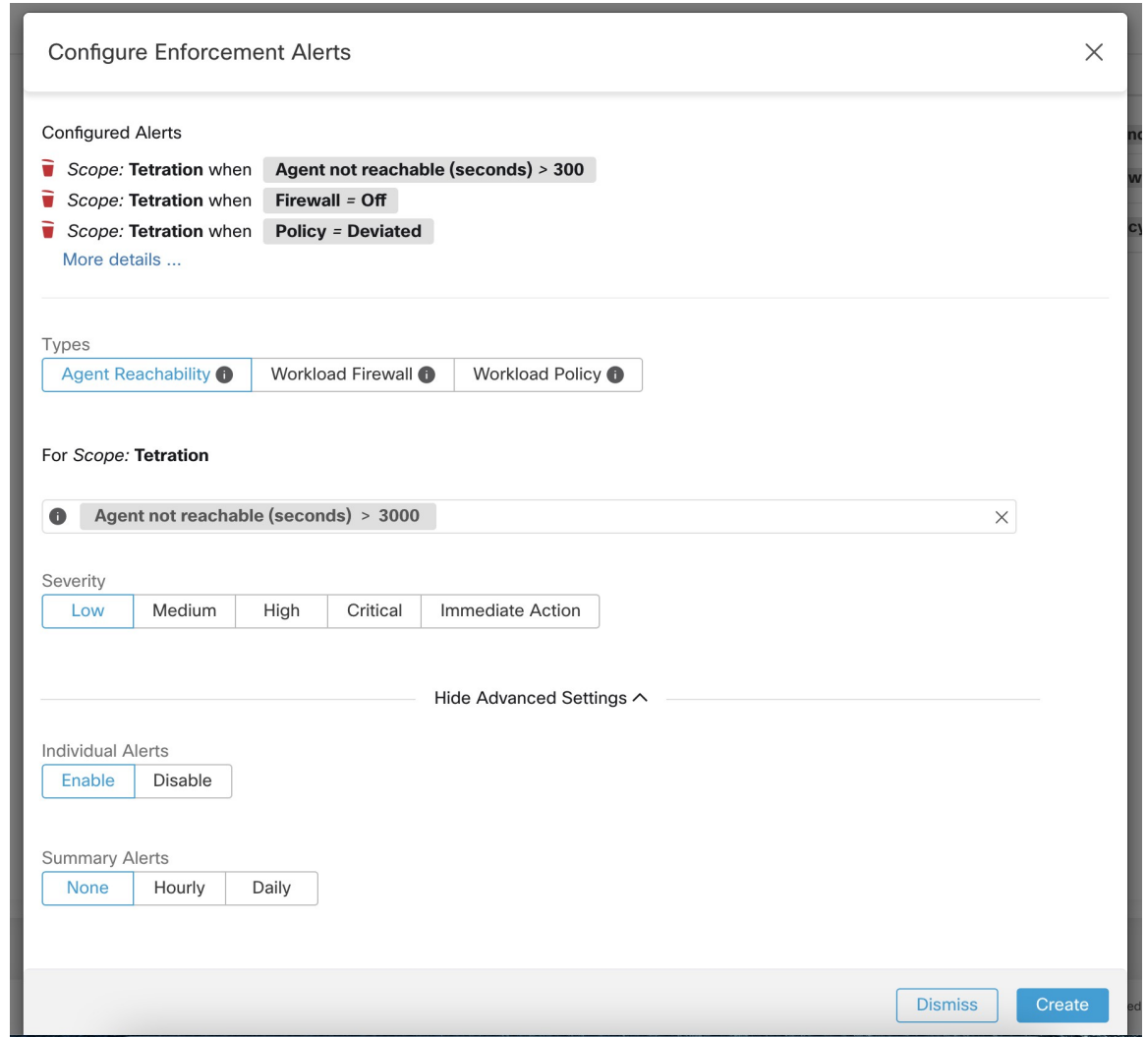


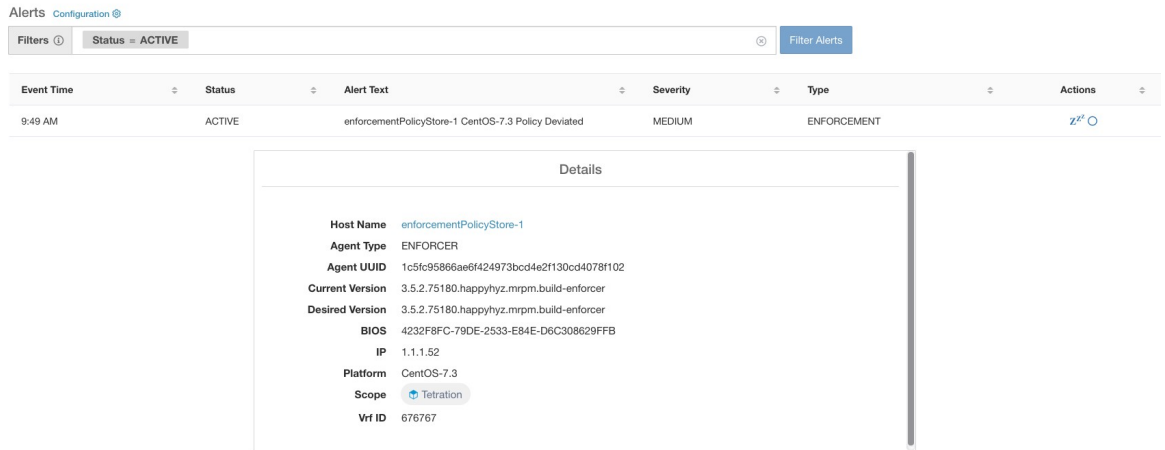
Figure 20: 在警报配置页面上查看已配置的执行警报

Alerts Trigger Rules

Alert Type ↑↓	Configuration ↑↓	Actions ↓
ENFORCEMENT	Scope: Tetration when Agent not reachable (seconds) > 300	
ENFORCEMENT	Scope: Tetration when Firewall = Off	
ENFORCEMENT	Scope: Tetration when Policy = Deviated	

执行 UI 警报详细信息

Figure 21: 执行警报详细信息



Alerts Configuration

Filters Status = ACTIVE Filter Alerts

Event Time	Status	Alert Text	Severity	Type	Actions
9:49 AM	ACTIVE	enforcementPolicyStore-1 CentOS-7.3 Policy Deviated	MEDIUM	ENFORCEMENT	Z ² ○

Details

Host Name enforcementPolicyStore-1

Agent Type ENFORCER

Agent UUID 1c5fc95866ae6424973bcd4e21130cd4078f102

Current Version 3.5.2.75180.happyhyz.mrpm.build-enforcer

Desired Version 3.5.2.75180.happyhyz.mrpm.build-enforcer

BIOS 4232F8FC-79DE-2533-E84E-D6C308629FFB

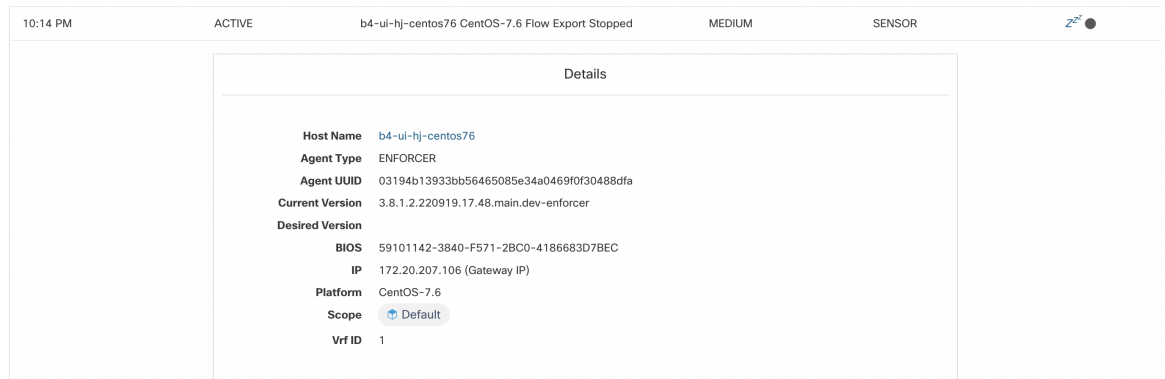
IP 1.1.1.52

Platform CentOS-7.3

Scope Tetration

Vrf ID 676767

Figure 22: 在主机上启用代理时的执行警报详细信息



10:14 PM ACTIVE b4-ui-hj-centos76 CentOS-7.6 Flow Export Stopped MEDIUM SENSOR Z² ●

Details

Host Name b4-ui-hj-centos76

Agent Type ENFORCER

Agent UUID 03194b13933bb56465085e34a0469f0f30488dfa

Current Version 3.8.1.2.220919.17.48.main.dev-enforcer

Desired Version 59101142-3840-F571-2BC0-4186683D7BEC

IP 172.20.207.106 (Gateway IP)

Platform CentOS-7.6

Scope Default

Vrf ID 1

执行警报详细信息

有关一般警报结构和有关字段的信息，请参阅[通用警报结构](#)。`alert_details` 字段是结构化的，包含以下用于执行警报的子字段

字段	警报类型	格式	说明
AgentType	<i>all</i>	字符串	“ENFORCER” 或 “SENSOR”，具体取决于安装类型
HostName	<i>all</i>	字符串	部署代理的主机名
IP	<i>all</i>	字符串	节点/网关的 IP 地址
Bios	<i>all</i>	字符串	节点的 BIOS UUID

字段	警报类型	格式	说明
平台	<i>all</i>	字符串	节点的平台/操作系统信息
CurrentVersion	<i>all</i>	字符串	节点上代理的软件版本
DesiredVersion	<i>all</i>	字符串	代理所需的软件版本
LastConfigFetchAt	<i>all</i>	整数	代理上次发送 <code>https</code> 请求的 Unix 时间戳

执行警报的 `alert_details` 示例

```
{
  "AgentType": "ENFORCER",
  "Bios": "72EF1142-03A2-03BC-C2F8-F600567BA320",
  "CurrentVersion": "3.5.1.1.mrpm.build.win64-enforcer",
  "DesiredVersion": "",
  "HostName": "win2k12-production-db",
  "IP": "172.26.231.193",
  "Platform": "MSServer2012R2Standard"
}
```

传感器警报

传感器警报配置提供了配置不同类型警报的功能，您可以设置警报的严重程度和配置参数的类型。

有关详细信息，请参阅[警报配置模式](#)。



Note 从 Cisco Secure Workload 3.5 开始，您可以使用警报配置模型来配置传感器警报。

Figure 23: 配置传感器警报

Configure Sensors Alerts [X]

Configured Alerts

- Scope: **Default** when **Agent Upgrade Status = Failed**
- Scope: **Default** when **Agent Flow Export Status = Stopped**
- Scope: **Default** when **Agent Check-In Service = Inactive**

[More details ...](#)

Types Agent Upgrade (i) Agent Flow Export (i) Agent Check In (i)

For Scope: **Default**

When (i) Agent Upgrade Status is Failed (x)

Severity Low Medium High Critical Immediate Action

Hide Advanced Settings ^

Individual Alerts Enable Disable

Summary Alerts None Hourly Daily

Create Dismiss

Configure Sensors Alerts [X]

Configured Alerts

- Scope: Default when **Agent Upgrade Status = Failed**
- Scope: Default when **Agent Flow Export Status = Stopped**
- Scope: Default when **Agent Check-In Service = Inactive**

[More details ...](#)

Types Agent Upgrade ⓘ Agent Flow Export ⓘ Agent Check In ⓘ

For Scope: Default

When ⓘ Agent Upgrade Status is Failed ⓘ

Severity Low Medium High Critical Immediate Action

Hide Advanced Settings ^

Individual Alerts Enable Disable

Summary Alerts None Hourly Daily

Create **Dismiss**

配置传感器警报，以便在代理升级失败时进行报告。如果代理无法升级到所需版本，则会触发此警报。

Configure Sensors Alerts [X]

Configured Alerts

- Scope: **Default** when **Agent Upgrade Status = Failed**
- Scope: **Default** when **Agent Flow Export Status = Stopped**
- Scope: **Default** when **Agent Check-In Service = Inactive**

[More details ...](#)

Types Agent Upgrade ⓘ Agent Flow Export ⓘ Agent Check In ⓘ

For Scope: **Default**

When ⓘ Agent Flow Export Status is Stopped ⓘ

Severity Low Medium High Critical Immediate Action

Hide Advanced Settings ^

Individual Alerts Enable Disable

Summary Alerts None Hourly Daily

Create **Dismiss**

配置传感器警报，以检测何时必须停止代理流导出。如果代理与集群之间的连接受阻，从而无法发送或交付流量和其他系统信息，则会触发此警报。

Configure Sensors Alerts

Configured Alerts

- Scope: Default when Agent Upgrade Status = Failed
- Scope: Default when Agent Flow Export Status = Stopped
- Scope: Default when Agent Check-In Service = Inactive

More details ...

Types Agent Upgrade Agent Flow Export **Agent Check In**

For Scope: Default

When Agent Check-In Service is Inactive

Severity Low Medium High Critical Immediate Action

Hide Advanced Settings ^

Individual Alerts Enable Disable

Summary Alerts None Hourly Daily

Create Dismiss

配置传感器警报，以检测代理 check_in 超时的情况。如果集群在超过 90 分钟后仍未收到代理的签入请求，则会触发此警报。

Figure 24: 在警报配置中配置传感器警报

Alerts Trigger Rules

Filters Alert type = sensors Filter Alerts

Alert Type	Configuration	Actions
SENSORS	Scope: Default when Agent Upgrade Status = Failed	
SENSORS	Scope: Default when Agent Flow Export Status = Stopped	
SENSORS	Scope: Default when Agent Check-In Service = Inactive	

传感器 UI 警报详细信息

Figure 25: 传感器警报

The screenshot shows the Alerts Configuration interface. At the top, there are filters for 'Status = ACTIVE' and a 'Filter Alerts' button. Below this is a table with columns: Event Time, Status, Alert Text, Severity, Type, and Actions. A single alert is listed with the following details:

Event Time	Status	Alert Text	Severity	Type	Actions
11:13 AM	ACTIVE	b4-ui-centos76 CentOS-7.6 Agent Inactive	MEDIUM	SENSOR	z''

Below the table is a 'Details' section for the selected alert, showing the following information:

- Host Name: b4-ui-centos76
- Agent Type: ENFORCER
- Agent UUID: c6c2fbed5e510f5f4eb43b98d30add8ab3fd907
- Current Version: 3.6.1.2.201213.21.41.main.dev-enforcer
- Desired Version: 3.6.1.2.201213.21.41.main.dev-enforcer
- BIOS: 59101142-3840-F571-2BC0-4186683D7BEC
- IP: 172.20.207.106
- Platform: CentOS-7.6
- Scope: Default
- Vrf ID: 1

传感器警报详细信息

有关警报的一般结构和有关字段的信息，请参阅“通用警报结构”。`alert_details` 字段是结构化的，包含用于传感器警报的以下子字段

字段	警报类型	格式	说明
AgentType	<i>all</i>	字符串	ENFORCER 或 SENSOR，具体取决于安装类型
HostName	<i>all</i>	字符串	部署代理的主机名
IP	<i>all</i>	字符串	节点/网关的 IP 地址
Bios	<i>all</i>	字符串	节点的 BIOS UUID
Platform	<i>all</i>	字符串	节点的平台/操作系统信息
CurrentVersion	<i>all</i>	字符串	节点上代理的软件版本
DesiredVersion	<i>all</i>	字符串	代理所需的软件版本
LastConfigFetchAt	<i>all</i>	整数	代理上次发送 HTTPS 请求的 Unix 时间戳

传感器警报的 `alert_details` 示例

```
{
  "AgentType": "SENSOR",
  "Bios": "72EF1142-03A2-03BC-C2F8-F600567BA320",
  "CurrentVersion": "3.5.1.1.mrpm.build.win64-sensor",
  "DesiredVersion": "",
  "HostName": "win2k12-production-db",
  "IP": "172.26.231.193",
  "Platform": "MSServer2012R2Standard"
}
```

常见问题解答

本部分列出了在部署和运行软件代理期间可能会遇到的一些问题。

概述

日志文件：日志文件存储在 `<install-location>/logs` 或 `<install-location>/log` 文件夹。日志文件通过 Cisco Secure Workload 服务进行监控和轮换。

代理部署

Linux

问：当命令无法安装代理并显示以下错误时，

```
rpm -Uvh tet-sensor-1.101.2-1.el6-dev.x86_64.rpm
```

我该怎么办：

```
error: cannot create transaction lock on /var/lib/rpm/.rpm.lock (Permission denied).
```

答：如果您没有安装代理的适当权限，请切换到根权限或使用 `sudo` 来安装代理。

问：运行 “`sudo rpm -Uvh tet-sensor-1.0.0-121.1b1bb546.el6-dev.x86_64.rpm`” 并遇到以下错误时，会发生什么情况：

```
Preparing... ##### [100%]
which: no lsb_release in (/sbin:/bin:/usr/sbin:/usr/bin:/usr/X11R6/bin)
error: %pre(tet-sensor-site-1.0.0-121.1b1bb546.x86_64) scriptlet failed, exit status 1
error: install: %pre scriptlet failed (2), skipping tet-sensor-site-1.0.0-121.1b1bb546
```

答：系统不符合安装代理的要求。在这种特定情况下，未安装 `lsb_release` 工具。

有关详细信息，请参阅软件代理部署标签部分，并安装所需的依赖关系。

问：运行 “`sudo rpm -Uvh tet-sensor-1.0.0-121.1b1bb546.el6-dev.x86_64.rpm`” 并遇到以下错误时，会发生什么情况：

```

Unsupported OS openSUSE project
error: %pre(tet-sensor-1.101.1-1.x86_64) scriptlet failed, exit status 1
error: tet-sensor-1.101.1-1.x86_64: install failed
warning: %post(tet-sensor-site-1.101.1-1.x86_64) scriptlet failed, exit status 1

```

答：您的操作系统不支持运行软件代理（在此特定情况下，“openSUSE 项目”是不支持的平台）。有关详细信息，请参阅“软件代理部署标签”部分。

问：在我安装了所有依赖关系并以适当权限运行安装程序后，没有出现任何错误。如何知道代理已安装成功？

答：要在安装代理后验证是否安装成功，请运行以下命令：

```

$ ps -ef | grep -e csw-agent -e tet-
root      14158      1  0 Apr03 ?        00:00:00 csw-agent
root      14160 14158  0 Apr03 ?        00:00:00 csw-agent watch_files
root      14161 14158  0 Apr03 ?        00:00:03 csw-agent check_conf
root      14162 14158  0 Apr03 ?        00:01:03 tet-sensor -f conf/.sensor_config
root      14163 14158  0 Apr03 ?        00:02:38 tet-main --sensoridfile=./sensor_id
root      14164 14158  0 Apr03 ?        00:00:22 tet-enforcer --logtostderr
tet-sen+  14173 14164  0 Apr03 ?        00:00:21 tet-enforcer --logtostderr
tet-sen+  14192 14162  0 Apr03 ?        00:07:23 tet-sensor -f conf/.sensor_config

```

您必须看到 *csw-agent* 的三个条目，以及 *tet-sensor* 的至少两个条目。如果服务未运行，请确保以下目录可用，否则表明安装失败。

- /usr/local/tet（适用于大多数 Linux 发行版）
- /opt/cisco/tetration（适用于 AIX、Ubuntu）
- /opt/cisco/secure-workload（适用于 Solaris、Debian）

Windows 的 ISE 安全评估代理

问：运行 PowerShell 代理安装程序脚本时，我收到以下错误之一：

1. 基础连接已关闭：接收时发生意外错误。
2. 客户端和服务器无法通信，因为它们没有共同的算法

答：这很可能是因为主机和服务器配置的 SSL/TLS 协议不匹配所致。可以使用以下命令检查 SSL/TLS 版本：

```
[Net.ServicePointManager]::SecurityProtocol
```

要设置与服务器匹配的 SSL/TLS，可以使用以下命令（请注意，这并非永久性更改，只是当前 PowerShell 会话的临时更改）：

```
[Net.ServicePointManager]::SecurityProtocol =
[System.Net.SecurityProtocolType]' Ssl3,Tls,Tls11,Tls12'
```

问：从下载的捆绑包运行 MSI 安装程序时，收到以下错误：

```
This installation package could not be opened. Verify that the package exists and that you
```


can access it, or contact the application vendor to verify that this is a valid Windows Installer package.

答：确保 `C:\Windows\Installer` 路径存在。如果是从命令行运行 MSI 安装程序，请确保在指向 MSI 文件时不包含相对路径。正确语法示例：

```
msiexec /i "TetrationAgentInstaller.msi" /l*v "msi_install.log" /norestart
```

问：我发现如果底层 NIC 是 Nutanix VirtIO 网络驱动程序，则 Windows 传感器软件无法升级。

答：在启用了接收分段合并的情况下，Npcap 0.9990 与 Nutanix VirtIO 网络驱动程序 1.1.3 之前的版本之间存在不兼容问题。

此问题的解决方法是将 Nutanix VirtIO 网络驱动程序升级到版本 1.1.3 或更高版本。

问：我安装了 Windows 传感器。传感器似乎未注册，并且 `sensor_id` 文件包含以下内容：
`uuid-invalid-platform`

答：对于 Windows，PATH 变量中可能没有 `system32`。检查 PATH 中是否有 `system32`，如果没有，则运行以下命令：

```
set PATH=%PATH%;C:\Windows\System32\
```

问：在 Windows 节点上，我没有收到来自 Kubernetes Pod 的网络流。

答：要验证所需的会话是否正在运行，以便从 Windows 节点上的 Kubernetes Pod 捕获流，请执行以下操作：

1. 使用管理权限运行 `cmd.exe`。
2. 运行以下命令：`logman query -ets`

确保以下会话正在运行：

- `CSW_MonNet`: 捕获网络流
- `CSW_MonHCS`: 监控 Pod 的创建
- `CSW_MonNat`: 监控 NAT 的流

Kubernetes

如果在 Kubernetes 守护进程集安装期间安装程序脚本失败，可能有很多原因。

问：是否可以从节点访问 Docker 注册表服务映像？

答：调试从思科 Cisco Secure Workload 集群提取映像的集群的直接或 HTTPS 代理问题

问：容器运行时是否会报告 SSL/TLS 不安全错误？

答：验证所有 Kubernetes 节点上是否已在容器运行时的适当位置安装了 Cisco Secure Workload HTTPS CA 证书。

问：映像下载失败的 Docker 注册表身份验证和授权？

答：在每个节点上，尝试使用 Helm 图表创建的密钥中的 Docker 拉取密钥手动从守护进程集规范中的注册表 URL 提取映像。如果手动映像提取也失败，则需要从 Cisco Secure Workload 集群注册表身份验证服务提取日志，以进一步调试问题。

问：在 Cisco Secure Workload 设备内部托管的 Kubernetes 集群是否正常？

答：检查集群的服务状态页面，确保所有相关服务均正常运行。从探索页面运行 dstool 快照并检索生成的日志。

问：Docker 映像生成器后台守护程序是否正在运行？

答：从 dstool 日志验证构建后台守护程序是否正在运行。

问：构建 Docker 映像的作业是否会失败？

答：从 dstool 日志验证是否未构建映像。Docker 构建 Pod 日志可用于调试构建工具包构建过程中的错误。执行协调器日志还可用于进一步调试构建失败。

问：创建 Helm 图表的作业是否失败？

答：从 dstool 日志验证是否未构建 Helm 图表。执行协调器日志将包含 Helm 构建作业的输出，可用于调试 Helm 图表构建作业失败的确切原因。

问：安装 bash 脚本已损坏？

答：尝试再次下载安装 bash 脚本。bash 脚本包含其附加的二进制文件数据。如果使用文本编辑器以任何方式编辑 bash 脚本或将其保存为文本文件，二进制文件数据中的特殊字符可能会被文本编辑器混淆或修改。

问：Kubernetes 集群配置 - 变体和风格过多，我们支持经典 K8。

答：如果客户运行的是 Kubernetes 的变体，则在部署的不同阶段可能存在许多故障模式。对故障阶段进行分类 - kubectl 命令运行失败、Helm 命令运行失败、Pod 映像下载失败、Pod 特权模式选项被拒绝、Pod 映像信任内容签名失败、Pod 映像安全扫描失败、Pod 二进制文件无法运行（架构不匹配）、Pod 运行，但 Cisco Secure Workload 服务无法启动，Cisco Secure Workload 服务由于异常的操作环境而启动，但出现运行时错误。

问：Kubernetes RBAC 凭证是否失效？

答：为了运行特权守护进程集，我们需要 K8s 集群的管理员权限。验证 kubectl 配置文件的默认情景是否指向目标集群和该集群的管理员等效用户。

问：Busybox 映像是否可从所有集群节点获取或下载？

答：修复连接问题，并手动测试是否可以下载 Busybox 映像。Pod 规范中使用的 busybox 版本必须在所有集群节点上可用（预先设定种子）或可下载。

问：安装过程中出现 API 服务器和 etcd 错误或一般超时？

答：由于 Kubernetes 集群中所有节点上的守护进程集 Pod 实例化，集群上的 CPU/磁盘/网络负载可能会突然激增。这在很大程度上取决于客户的特定安装详细信息。由于过载，安装过程（在所有节点上提取并写入磁盘的映像）可能需要太长时间，或者 Kubernetes API 服务器或 Cisco Secure Workload Docker 注册表终端或（如已配置）代理服务器临时过载。短暂等待所有节点上的镜像提取完成，Kubernetes 集群节点上的 CPU/磁盘/网络负载减少后，再次重试安装脚本。来自 Kubernetes 控制平面

的 API Server 和 etcd 错误表明，Kubernetes 控制平面节点可能配置不足，或受到活动突然激增的影响。

问：Cisco Secure Workload 代理在运行时遇到运行时问题？

答：如果 Pod 已正确部署且代理已开始运行但遇到运行时问题，请参阅“Linux 代理故障排除”部分。Kubernetes 部署成功安装并启动 Pod 后，故障排除步骤相同。

异常类型

以下是使用和管理 Cisco Secure Workload 代理时工作流程中最常见的问题。

代理不活动

代理已停止检查集群服务。出现这种情况有多种原因：

- 主机可能已关闭
- 网络连接已被防火墙规则中断或阻止
- 该代理服务已停止

所有平台

- 验证主机是否处于活动状态且运行正常
- 验证代理服务已启动且正在运行
- 验证与集群的网络连接是否正常

升级失败

代理升级失败。这可能由少数情况触发，例如：

- 签入脚本尝试下载软件包时找不到软件包 - 无法解压升级软件包或无法验证软件包中的安装程序。
- 安装过程因操作系统问题或依赖关系而失败。

Windows 的 ISE 安全评估代理

- 缺少 CA 根证书：[证书问题](#)
- 如果代理最初是使用 MSI 安装软件包手动安装的，请检查 Windows 版本是否与用户指南中支持的[平台列表](#)匹配：[检查当前是否支持平台](#)
- 检查以确保已为 Windows Installer 操作正确配置操作系统：[Windows 安装程序问题](#)
- 确保主机上有足够的可用磁盘空间

Linux

- 如果自上次安装代理以来已升级主机操作系统，请验证当前版本是否与用户指南中支持的平台列表匹配：[检查当前是否支持平台](#)
- 确保自上次安装以来所需的依赖关系未发生任何更改。您可以使用 `-no-install` 选项来运行代理安装程序脚本，以重新验证这些依赖关系。
- 确保主机上有足够的可用磁盘空间

AIX

- 确保自上次安装以来所需的依赖关系未发生任何更改。您可以使用 `-no-install` 选项来运行代理安装程序脚本，以重新验证这些依赖关系。
- 确保主机上有足够的可用磁盘空间

转换失败

当前代理类型与所需的代理类型不匹配，并且转换尝试已超时。当代理执行 `check_in` 下载软件包或 `wss` 服务未能向代理推送 `convert_commnad` 时，可能就会出现通信问题。

所有平台

- 验证当前版本和代理类型是否与用户指南中支持的平台列表匹配：[检查当前是否支持平台](#)

转换功能

并非所有代理都支持将代理从一种类型（例如深度可视性）转换为另一种类型（例如执行）。如果无法执行转换的代理需要转换，则会报告异常。

策略未同步

代理上次报告的当前策略 (NPC) 版本与集群上生成的当前版本不一致。造成这种情况的原因可能是代理和集群之间的通信错误、代理未能通过本地防火墙执行策略或代理执行服务未运行。

Windows 的 ISE 安全评估代理

- 如果执行模式为 WAF，请验证主机上不存在会阻止防火墙启用的 GPO，从而添加规则（关闭“保留规则”）或设置默认操作：[GPO 配置](#)
- 验证主机和集群之间是否存在连接：[SSL 故障排除](#)
- 验证生成的规则计数是否小于 **2000**
- 验证 `WindowsAgentEngine` 服务是否正在运行：`sc query windowsagentengine`
- 验证是否有可用的系统资源

Linux

- 使用 `iptables` 和 `ipset` 命令来验证 `iptables` 和 `ipset` 是否存在
- 验证主机和集群之间是否存在连接: [SSL 故障排除](#)
- 验证 `tet-enforcer` 进程是否正在运行: `ps -ef | grep tet-enforcer`

AIX

- 使用 `ipf -V` 命令验证 `ipfilter` 是否已安装并正在运行
- 验证主机和集群之间是否存在连接: [SSL 故障排除](#)
- 验证 `tet-enforcer` 进程是否正在运行: `ps -ef | grep tet-enforcer`

流导出: Pcap 打开

如果 Cisco Secure Workload 代理无法打开 pcap 设备来捕获流, 则您会在代理日志中看到错误。成功打开 Pcap 设备将报告如下:

Windows 日志: `C:\Program Files\Cisco Tetration\Logs\TetSen.exe.log`

```
I0609 15:25:52.354 24248 Started capture thread for device <device_name>  
I0609 15:25:52.354 71912 Opening device {<device_id>}
```

Linux 日志: `/usr/local/tet/logs/tet-sensor.log`

```
I0610 03:24:22.354 16614 Opening device <device_name>  
[2020/06/10 03:24:23:3524] NOTICE: lws_client_connect_2: <device_id>: address 172.29.  
→136.139
```

流导出: HTTPS 连接

代理与集群之间的连接在外部被阻止, 因此无法传送流和其他系统信息。这是由于网络防火墙、SSL 解密服务或主机上的第三方安全代理存在一个或多个配置问题造成的。

- 如果代理和集群之间存在已知的防火墙或 SSL 解密安全设备, 请确保允许与所有 Cisco Secure Workload 收集器和 VIP IP 地址的通信。对于本地集群, 收集器列表将在 Cisco Secure Workload Web 界面左侧导航栏中的故障排除 (Troubleshoot) > 虚拟机 (Virtual Machines) 下列出。查找 `collectorDatamover-*`。对于 Cisco Secure Workload 云, 您的门户中将列出所有需要允许的 IP 地址。
- 为了帮助确定是否存在 SSL 解密, 可使用 `openssl s_client` 来建立连接并显示返回的证书。添加到证书链中的任何其他证书都将被代理的本地 CA 拒绝。 [SSL 故障排除](#)

证书问题

Windows 的 ISE 安全评估代理

MSI 安装程序的证书问题

MSI 安装程序使用代码签名证书进行签名：

对于 MSI 安装程序，版本 3.6.x 及更高版本和 3.5.1.31 及更高版本

- 分支证书：Cisco Systems, Inc
- 中间证书：DigiCert Trusted G4 Code Signing RSA4096 SHA384 2021 CA1
- 根证书：DigiCert Trusted Root G4

对于 MSI 安装程序，早期版本

- 分支证书：Cisco Systems, Inc
- 中间证书：Symantec Class 3 SHA256 Code Signing CA
- 根证书：VeriSign Class 3 Public Primary Certification Authority - G5

它使用时间戳证书：

对于 MSI 安装程序，版本 3.6.x 及更高版本和 3.5.1.31 及更高版本

- 分支证书：Symantec SHA256 TimeStamping Signer - G3
- 中间证书：Symantec SHA256 TimeStamping CA
- 根证书：VeriSign Universal Root Certification Authority

对于 MSI 安装程序，早期版本

- 分支证书：Symantec SHA256 Timestamping Signer - G2
- 中间证书：Symantec SHA256 Timestamping CA
- 根证书：VeriSign Universal Root Certification Authority

如果 MSI 安装程序的数字签名无效，则 Windows 传感器安装或升级将失败。以下情况时数字签名无效：

- MSI 安装程序签名根证书或 MSI 安装程序时间戳根证书不在“受信任的根证书颁发机构”存储区内
- MSI 安装程序签名根证书或 MSI 安装程序时间戳根证书过期或被撤销。

问题 1

代理安装可能会失败，TetUpdate.exe.log 中会出现以下错误：“Msi signature is not trusted. 0x800b0109”

解决方法

- 从命令提示符运行命令 *certmgr*
- 检查 MSI 安装程序签名根证书或 MSI 安装程序时间戳根证书是否在不受信任的证书 (*Untrusted Certificates*) 存储区中。
- 将其移至受信任的根证书颁发机构 (*Trusted Root Certification Authority*) 存储区。

问题 2

Windows 传感器升级失败，TetUpdate.exe.log 中出现以下错误：“Msi signature is not trusted. 0x800B010C”

证书已被其颁发者明确撤销。

解决方法

- 从命令提示符运行命令 *certmgr*
- 检查 MSI 安装程序签名根证书或 MSI 安装程序时间戳根证书是否在不受信任的证书 (*Untrusted Certificates*) 存储区中。
- 将其复制到受信任的根证书颁发机构 (*Trusted Root Certification Authority*) 存储区。

问题 3

Windows 传感器升级失败，TetUpdate.exe.log 中出现以下内容：“Msi signature is not trusted. 0x80096005”

解决方法

- 从命令提示符运行命令 *certmgr*
- 检查 MSI 安装程序签名根证书或 MSI 安装程序时间戳根证书是否在“受信任的根证书颁发机构”存储区内

如果证书缺失，请从其他计算机导入证书。

要导入证书，请执行以下步骤：

首先从其中一个工作服务器导出 VeriSign 通用根证书颁发机构的证书。请按照以下步骤操作：

- 从命令提示符运行命令 *certmgr*
- 右键单击“受信任的根证书颁发机构” (Trusted Root Certification Authorities) 下的“VeriSign 通用根证书颁发机构” (VeriSign Universal Root Certification Authority) 证书，然后转到“所有任务导出” (All tasksExport)。
- 将导出的证书复制到非工作服务器，然后导入证书。

要导入证书，请执行以下步骤：

首先从其中一个工作服务器导出 VeriSign 通用根证书颁发机构的证书。请按照以下步骤操作：

- 从命令提示符运行命令 *certmgr*
- 右键点击“受信任的根证书颁发机构” (Trusted Root Certification Authorities) 下的证书选项卡，然后转到“所有任务导入” (All tasks Import)。
- 选择复制的根证书并将其添加到存储库中。

NPCAP 安装程序的证书问题

适用于 **Windows 2012、Windows 2012 R2、Windows 8、Windows 8.1**

NPCAP 版本：1.55

NPCAP 签名证书：

- 分支证书：Insecure.Com LLC
- 中间证书：DigiCert EV Code Signing CA (SHA2)
- 根证书：DigiCert High Assurance EV Root CA

NPCAP 时间戳证书：

- 分支证书：DigiCert Timestamp 2021
- 中间证书：DigiCert SHA2 Assured ID Timestamping CA
- 根证书：DigiCert Assured ID Root CA

问题 1

Windows 代理安装可能会失败，*msi_installer.log* 中会出现以下错误

```
CheckServiceStatus : Exception System.InvalidOperationException: Service npcap was not found
on computer
'. ' . -> System.ComponentModel.Win32Exception: The specified service does not exist as an
installed service
```

解决方法

- 从命令提示符运行命令 *certmgr*
- 选中“受信任的根证书颁发机构” (Trusted Root Certification Authority) 存储区中的“DigiCert High Assurance EV Root CA”。
- 如果证书缺失，请从其他计算机导入证书。

要导入证书，请执行以下步骤：

首先从其中一个工作服务器导出证书“DigiCert High Assurance EV Root CA”。请按照以下步骤操作：

- 从命令提示符运行命令 `certmgr`
- 右键点击“受信任的根证书颁发机构”下的证书“DigiCert High Assurance EV Root CA”，然后转到所有任务导出。
- 将导出的证书复制到非工作服务器，然后导入证书。

要导入证书，请执行以下步骤：

- 从命令提示符运行命令 `certmgr`
- 右键点击“受信任的根证书颁发机构”(Trusted Root Certification Authorities) 下的证书选项卡，然后转到“所有任务导入”(All tasksImport)。
- 选择复制的根证书并将其添加到存储库中。

适用于 Windows 2008 R2

NPCAP 版本：0.991

NPCAP 签名证书：

- 分支证书：Insecure.Com LLC
- 中间证书：DigiCert EV Code Signing CA
- 根证书：DigiCert High Assurance EV Root CA

NPCAP 时间戳证书：

- 分支证书：DigiCert Timestamp Responder
- 中间证书：DigiCert Assured ID CA-1
- 根证书：VeriSign DigiCert Assured ID Root CA

问题 1

Windows 代理安装可能会失败，`msi_installer.log` 中会出现以下错误

```
CheckServiceStatus : Exception System.InvalidOperationException: Service npcap was not found
on
computer '.' . -> System.ComponentModel.Win32Exception: The specified service does not
exist as an
installed service
```

解决方法

- 从命令提示符运行命令 `certmgr`
- 选中受信任的根证书颁发机构 (Trusted Root Certification Authority) 存储区中的 *DigiCert High Assurance EV Root CA*。
- 如果证书缺失，请从其他计算机导入证书。

要导入证书，请执行以下步骤：

首先从其中一个工作服务器导出证书“DigiCert High Assurance EV Root CA”。请按照以下步骤操作：

- 从命令提示符运行命令 `certmgr`
- 右键点击“受信任的根证书颁发机构”下的证书“DigiCert High Assurance EV Root CA”，然后转到所有任务导出。
- 将导出的证书复制到非工作服务器，然后导入证书。

要导入证书，请执行以下步骤：

- 从命令提示符运行命令 `certmgr`
- 右键点击“受信任的根证书颁发机构”(Trusted Root Certification Authorities)下的证书选项卡，然后转到“所有任务导入”(All tasksImport)。
- 选择复制的根证书并将其添加到存储库中。

Windows 主机重命名

场景 1：重命名 Windows 主机后无法查看 IP 地址和 VRF 信息，解决该问题的步骤如下：

- 从 TaaS UI 中删除条目（使用缺少 IP 地址和 VRF 信息的新主机名）。
- 从 Windows 主机卸载“Cisco Cisco Secure Workload Agent”，并删除“Cisco Tetration”目录（该目录的路径通常为：“C:\Program Files\Cisco Tetration”）。
- 在 Windows 主机上安装“Cisco Cisco Secure Workload Agent”。

按照上述步骤，代理就能成功在 TaaS UI 上注册 IP 地址和 VRF 信息。

场景 2：计划中的 Windows 主机重命名（提前），要遵循的步骤：

- 从 Windows 主机卸载“Cisco Cisco Secure Workload Agent”，并删除“Cisco Tetration”目录（该目录的路径通常为：“C:\Program Files\Cisco Tetration”）。
- 重命名 Windows 主机并重启。
- 在 Windows 主机上安装“Cisco Cisco Secure Workload Agent”（使用新主机名）。

按照上述步骤进行计划的主机重命名后，代理就会在 TaaS UI 上用新的主机名注册。

检查当前是否支持平台

Windows 的 ISE 安全评估代理

- 运行命令 `winver.exe`

- 将此版本与此处列出的版本进行比较：[支持的平台和要求](#)

Linux

- 运行 `cat /etc/os-release`
- 将此版本与此处列出的版本进行比较：[支持的平台和要求](#)

AIX

- 运行命令 `uname -a`
- 注意：主要版本与次要版本是相反的

```
p7-ops2> # uname -a
AIX p7-ops2 1 7 00F8AF944C00
```

- 在本例中，主机名后的第一个数字是次版本，第二个数字是主版本，即 AIX 7.1 版。将此版本与此处列出的版本进行比较：[支持的平台和要求](#)

Windows 安装程序问题

- 确保存在 `C:\Windows\Installer` 目录。这不会显示在文件资源管理器中，最简单的验证方法是在 CMD 会话中并运行：`dir C:\Windows\Installer`
- 检查 `Windows Installer` 服务是否未被禁用。它必须设置为手动 (*Manual*)
- 检查 `Windows Installer` 是否未报告其他错误。检查 **Windows 日志 (Windows Logs) > 应用 (Application) > 源 (Source) > MsiInstaller** 下的 Windows 系统事件日志

必需的 Windows 服务

以下是在禁用时与代理安装问题相关联的服务列表。建议在深度可视性和执行代理的初始安装以及任何升级期间运行这些服务。

Table 9: 必需的 *Windows* 服务

服务	安装目的
设备设置管理器	用于安装 Npcap 过滤器驱动程序的设备驱动程序管理。
设备安装服务	也用于安装 Npcap 过滤器驱动程序。
Windows 安装程序	安装代理 MSI 软件包时需要。
Windows Firewall	对于 WAF 执行模式，此字段为必填项。
应用体验	用于确定系统上的功能可执行文件。



Note 应用体验服务仅适用于 Windows Server 2008、2008R2、2012、2012R2 和 Windows 7 版本。如果禁用，Npcap 安装过程中可能会出现文件锁，从而导致安装失败。

Npcap 问题

Npcap 是仅用于 Windows 代理的 pcap 工具。在代理服务启动十秒后，它将尝试安装 Npcap 或将其升级为支持的版本。如果 Npcap 服务无法安装或升级，代理将在接下来的 30 分钟内重试安装。在尝试 3 次失败后，如果有可用的版本，代理会尝试将 Npcap 回滚到之前支持的版本。之后，代理将不再尝试安装 Npcap。您可以检查 `C:\Program Files\Cisco Tetration\Logs\TetUpdate.exe.log` 和 `C:\Program Files\Cisco Tetration\Logs\npcap_install.log` 以确定错误。

Npcap 不会升级（手动或通过代理）

- 如果进程当前正在使用 Npcap 库，则 Npcap 有时会无法正确卸载。要对此进行检查，请运行以下命令：

```
PS C:\Program Files\Npcap> .\NPFInstall.exe -check_dll
WindowsSensor.exe, Wireshark.exe, dumpcap.exe
```

如果您看到列出的进程，则必须先将其停止，然后才能继续 Npcap 升级。如果没有进程在使用 Npcap，上述命令将仅显示 `<NULL>`

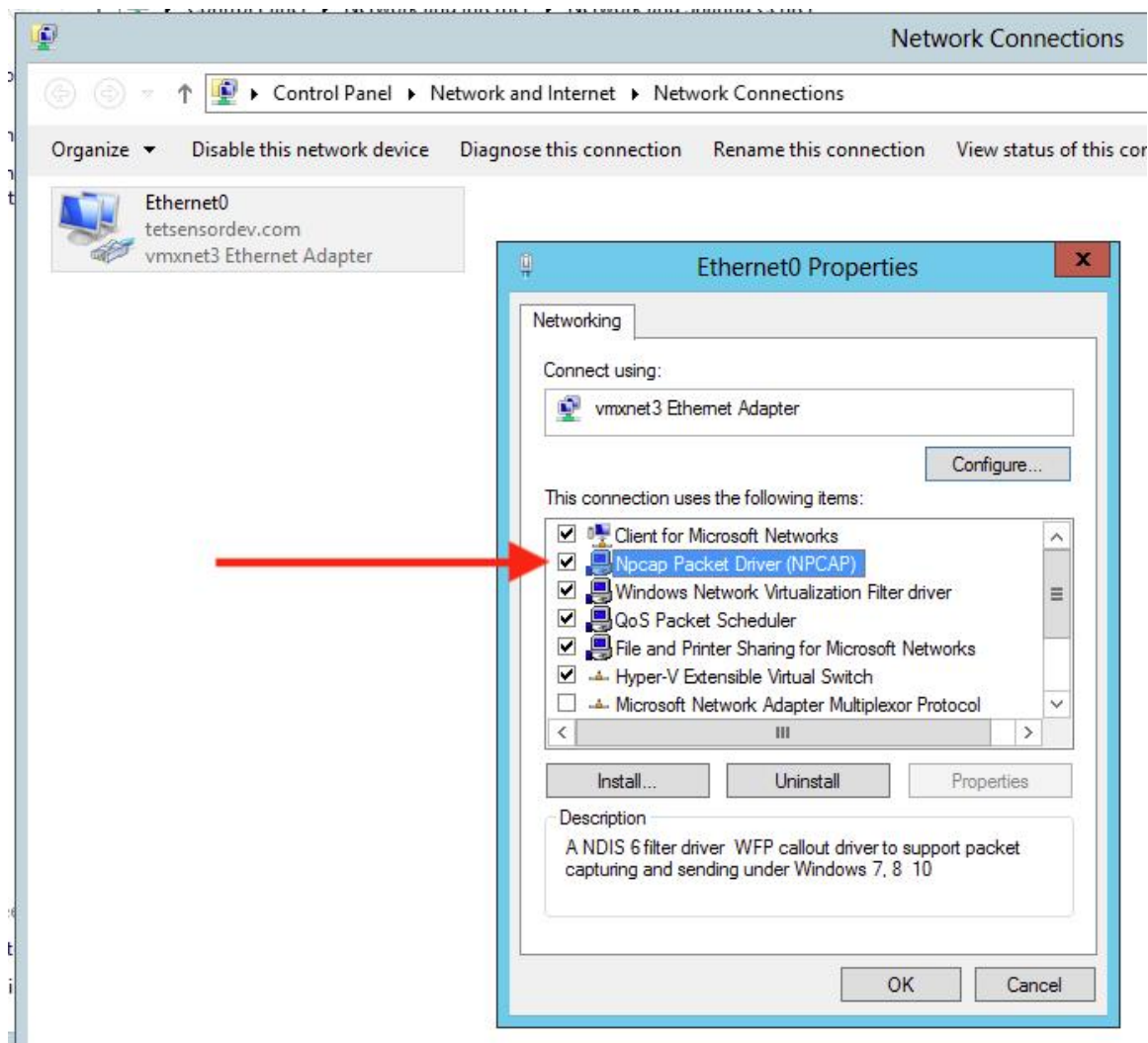
Npcap 不会安装

- 检查系统上安装的 CA 证书：[NPCAP 安装程序的证书问题](#)
- 检查 Windows 安装程序问题：[Windows 安装程序问题](#)
- 确认系统上没有其他用户正在对网络接口进行更改。否则可能会导致 COM 锁定，从而阻止 NDIS 驱动程序绑定。

验证 Npcap 是否已完全安装

Procedure

- 步骤 1** 检查控制面板 (Control Panel) > 程序 (Programs) 和功能 (Features)，以查看 Npcap 是否列为已安装的应用
- 步骤 2** 确保 Npcap 数据包驱动程序已绑定到相关 NIC（存在复选标记）



步骤 3 检查是否安装了网络驱动程序

```
C:\Windows\system32>pnputil -e | findstr Nmap
Driver package provider : Nmap Project
```

步骤 4 检查驱动程序服务是否已安装并正在运行

```
C:\Windows\system32>sc query npcap
SERVICE_NAME: npcap
        TYPE : 1 KERNEL_DRIVER
        STATE : 4 RUNNING
```

步骤 5 检查注册表项是否存在（由代理用于验证 Npcap 是否已存在）

```
C:\Windows\system32>reg query HKLM\software\wow6432node\npcap
HKEY_LOCAL_MACHINE\software\wow6432node\npcap
        AdminOnly REG_DWORD 0x1
        WinPcapCompatible REG_DWORD 0x0
        (Default) REG_SZ C:\Program Files\Npcap
```

步骤 6 检查已安装的 Npcap 程序文件是否齐全

```
C:\Windows\system32>dir "c:\program files\npcap"
Directory of c:\program files\npcap
04/29/2020 02:42 PM <DIR> .
04/29/2020 02:42 PM <DIR> ..
01/22/2019 08:16 AM 868 CheckStatus.bat
11/29/2016 03:43 PM 1,034 DiagReport.bat
12/04/2018 11:12 PM 8,908 DiagReport.ps1
01/09/2019 09:22 PM 2,959 FixInstall.bat
04/29/2020 02:42 PM 134,240 install.log
01/11/2019 08:52 AM 9,920 LICENSE
03/14/2019 08:59 PM 10,434 npcap.cat
03/14/2019 08:57 PM 8,657 npcap.inf
03/14/2019 09:00 PM 74,040 npcap.sys
03/14/2019 08:57 PM 2,404 npcap_wfp.inf
03/14/2019 09:00 PM 270,648 NPFInstall.exe
04/29/2020 02:42 PM 107,783 NPFInstall.log
03/14/2019 09:01 PM 175,024 Uninstall.exe
13 File(s) 806,919 bytes
2 Dir(s) 264,417,628,160 bytes free
```

步骤 7 检查 .sys 驱动程序文件是否位于 Windows 驱动程序文件夹中

```
C:\Windows\system32>dir "C:\Windows\System32\Drivers\npcap.sys"
Directory of C:\Windows\System32\Drivers
03/14/2019 09:00 PM 74,040 npcap.sys
1 File(s) 74,040 bytes
```

NPCAP 安装或升级期间的网络连接问题

仅适用于 Windows 2016

如果您在设置中具有第三方 LWF（轻量级过滤器）驱动程序（例如 netmon）或配置了组合适配器，并且在代理部署期间安装了 NPCAP，您可能会遇到

RDP 已重新连接

NetBIOS 服务已重启

类似网络连接问题

这是由 Windows 2016 操作系统中的 BUG 导致的

NIC 组合与 NPCAP 的兼容性问题

组合 NIC 功能基于底层物理 NIC（Intel、Broadcom、Realtek、MS 虚拟适配器等）和组合驱动程序配置（基于交换机、负载均衡或确保您的策略能解决不常见或不经常发生的活动和情况，如故障转移、从备份恢复故障转移、在多个 NIC 之间分发数据包的算法）。

某些 NPCAP 版本存在与组合 NIC 的兼容性问题，尤其是在绑定到下面的组合 NIC 期间。

当前使用的 Cisco Secure Workload 传感器软件已使用 Microsoft 支持的 NIC 组合进行测试。

```
NIC type : Intel(R) 82574L Gigabit Network Connection
Teaming Mode : Switch Independent
Load Balancing Mode: Address Hash
OS : Windows 2012 , Windows 2012 R2, Windows 2016, Windows 2019
NPCAP version: 1.55
```



Note Windows 2008R2 不支持 Microsoft 支持的 NIC 组合。

VDI 实例虚拟机不报告网络流

当 NPCAP 服务运行时，TetSensor 服务有时不会捕获克隆虚拟机上的网络流。如果使用 MSI 安装程序安装代理时没有 **nostart** 标志，或使用 PowerShell 安装程序在虚拟机模板或黄金镜像上安装代理时没有 **goldImage** 标志，就会出现这种情况。

在这种情况下，Cisco Secure Workload 代理服务将开始在 VM 模板上运行。已安装 NPCAP 并将其绑定到虚拟机模板上的网络堆栈。从虚拟机模板克隆新虚拟机时，NPCAP 未正确绑定到新克隆虚拟机上的网络堆栈。因此，NPCAP 无法捕获网络流。

使用 NPCAP 时的网络性能

据观察，当 Windows TetSensor 服务运行时，网络性能会受到影响。Windows Tet- 传感器服务 (tetsen.exe) 使用 NPCAP 捕获网络流。实施 NPCAP 以捕获网络流以及流向 tetsen.exe 的网络流会影响网络性能。

在安装 tetsensor 后比较网络性能，客户端：Windows 2016

NPCAP 1.55

TetSensor 配置：具有执行模式 WFP 的对话模式

服务器：Windows 2016

NPCAP 1.55

TetSensor 配置：具有执行模式 WFP 的对话模式

运行 cmd : iperf3.exe -c <server_ip> -t 40

Table 10: 121071: 使用 NPCAP 155 时的网络性能

设置	网络性能
未安装 TetSensor 无 NPCAP	[ID] 间隔传输带宽 [4] 0.00-40.00 秒 18.2 GB 3.90 Gbit/秒发送方 [4] 0.00-40.00 秒 18.2 GB 3.90 Gbits/秒接收方
已安装 TetSensor 已安装 NPCAP	[ID] 间隔传输带宽 [4] 0.00-40.00 秒 17.3 GBytes 3.72 Gbits/秒发送方 [4] 0.00-40.00 秒 17.3 GB 3.72 Gbits/秒接收方

使用 NPCAP 0.9990 时的网络性能

在安装 tetsensor 后比较网络性能，客户端：Windows 2016

NPCAP 0.9990

TetSensor 配置：具有执行模式 WFP 的对话模式

服务器：Windows 2016

NPCAP 0.9990

TetSensor 配置：具有执行模式 WFP 的对话模式

运行 cmd : iperf3.exe -c <server_ip> -t 40 .. table:: 使用 NPCAP 0.9990 时的网络性能

class 长表

设置	网络性能
已安装 TetSensor	[ID] 间隔传输带宽
已安装 NPCAP	[4] 0.00-40.00 秒 16.3 GB 3.50 Gbit/秒发送方 [4] 0.00-40.00 秒 16.3 GBytes 3.50 Gbits/秒接收方



Note 性能可能因安装的 Windows NPCAP 版本、Windows 操作系统和网络配置而异。

操作系统性能和/或稳定性问题

如果 Cisco Secure Workload 软件不支持安装的 NPCAP 版本或 NPCAP 配置，操作系统可能会遇到未知的性能或稳定性问题。

支持的 NPCAP 版本：0.991 和 1.55

GPO 配置

执行策略的代理仅需要使用本地设置或 GPO 启用防火墙。不应设置所有其他 GPO 设置，并将其保留为“未配置”(Not Configured)。

- 要检查 GPO 设置是否阻止执行，您可以检查 `C:\Program Files\Cisco Tetration\Logs\TetEnf.exe.log` 日志并搜索以下错误示例：
- 与“**Preserve Rules = No**”设置冲突的规则：“组策略中设置了防火墙规则。Cisco Secure Workload 代理无权删除这些”(There are firewall rules set in the Group Policy. Secure Workload agent does not have permission to remove these)
- 防火墙设置为关闭：“GPO 已为 DomainProfile 禁用防火墙”(GPO has disabled firewall for DomainProfile)
- 已设置默认操作：“组策略与 DomainProfile 的默认入站操作冲突”(Group Policy has conflicting default inbound action for DomainProfile)

- 要检查向主机应用了哪些 GPO 策略，请运行 `gpresult.exe /H gpresult.html` 并打开生成的 HTML 报告。在下面的示例中，如果“保留规则” (Preserve Rules) 设置为“否” (No)，则 *Cisco Secure Workload* 代理防火墙会应用将与执行冲突的入站规则。

The screenshot displays the Windows Firewall settings interface. A green box highlights the Firewall state settings, which are all set to 'Not Configured' except for 'Firewall state', which is 'On'. A green message states: 'Recommended Configuration Firewall state = On All other settings = Not Configured'. Below this, a red box highlights the 'Inbound Rules' section, which is labeled 'Inbound/Outbound Rules Not Recommended'. A table shows one rule: 'HTTP Inbound Rule' with a description 'This rule might contain some elements that cannot be interpreted by the current version of GPMC reporting' and a status of 'Enabled'.

Policy	Setting	Winning GPO
Firewall state	On	Tetration Agent Firewall
Inbound connections	Not Configured	
Outbound connections	Not Configured	
Apply local firewall rules	Not Configured	
Apply local connection security rules	Not Configured	
Display notifications	Not Configured	
Allow unicast responses	Not Configured	
Log dropped packets	Not Configured	
Log successful connections	Not Configured	
Log file path	Not Configured	
Log file maximum size (KB)	Not Configured	

Name	Description	Winning GPO
HTTP Inbound Rule	This rule might contain some elements that cannot be interpreted by the current version of GPMC reporting	Tetration Agent Firewall

代理到集群的通信

Cisco Secure Workload 代理通过多个信道来维护与集群的连接。根据代理的类型，连接数会有所不同。

连接类型

- **WSS:** 通过端口 443 与集群建立持久性套接字连接
- **签入:** 每 15-20 分钟对集群进行一次 HTTPS 调用，以检查当前配置、检查更新以及将代理的活动状态更新到集群。这也会报告升级失败。
- **流导出:** 通过端口 443 (TaaS) 或 5640 (内部部署) 建立持久性 SSL 连接，以将流元数据发送到集群
- **执行:** 通过端口 443 (Taas) 或 5660 (内部部署) 建立持久性 SSL 连接，以获取执行策略并报告执行状态

检查连接状态

迭代UI将报告代理处于非活动状态（不再签入）、没有导出的流（在“代理工作负载配置文件”（Agent Workload Profile）页面上的“统计信息”（Stats）下）或执行失败。根据错误的情况，您可以检查工作负载上的不同日志，以帮助确定问题的根源。

非活动代理

Windows 日志: *C:\Program Files\Cisco Tetration\Logs\TetUpdate.exe.log*

Linux 日志: */usr/local/tet/logs/check_conf_update.log*

预期 HTTP 响应代码为 304，表示没有配置更改。错误代码 = 2 也是预期值。任何其他 HTTP 响应代码都表示与 Cisco Secure Workload 集群上的 WSS 服务存在通信问题。

```
Tue 06/09/2020 17:25:25.08 check_conf_update: "curl did not return 200 code, it's 304,
→ exiting"
Tue 06/09/2020 17:25:25.08 check_conf_update: "error code after running check_conf_
→update = 2"
```

- **304** 预期值，无配置更改。签入成功
- **401** 注册不成功，缺少激活密钥 (TaaS)
- **403** 代理已注册到具有相同 UUID 的集群
- **000** 表示 SSL 连接存在问题。curl 无法访问 WSS 服务器，或者证书存在问题。请参阅 [SSL 故障排除](#)：[SSL 故障排除](#)

没有导出的流

Windows 日志: *C:\Program Files\Cisco Tetration\Logs\TetSen.exe.log*

Linux 日志: */usr/local/tet/logs/tet-sensor.log*

以下信息表示已成功连接到 WSS

```
cfgserver.go:261] config server: StateConnected, wss://<config_server_ip>:443/wss/
→<sensor_id>/forensic, proxy:
```

以下信息表示已成功连接到收集器

```
collector.go:258] next collector: StateConnected, ssl://<collector_ip>>:5640
```

如果连接到 WSS 或收集器时出错，请检查防火墙配置或验证代理与 Cisco Secure Workload 之间是否正在发生任何 SSL 解密。请参阅：[SSL 故障排除](#)

未能执行策略

Windows 日志: *C:\Program Files\Cisco Tetration\Logs\TetEnf.exe.log*

Linux 日志: */usr/local/tet/logs/tet-enforcer.log*

```
ssl_client.cpp:341] Successfully connected to EFE server
```

如果连接到 EFE 服务器时出错，请检查防火墙配置或验证代理与 Cisco Secure Workload 之间是否正在发生任何 SSL 解密。请参阅：[SSL 故障排除](#)

SSL 故障排除

代理通信概述

Cisco Secure Workload 代理使用 TLS 来保护与 Cisco Secure Workload 云 SaaS 服务器的 TCP 连接。这些连接分为三个不同的通道。

- 代理 -> 端口 TCP/443 (TLS) (sensorVIP) 上的思科 Cisco Secure Workload SaaS 控制通道
这是一个低容量控制通道，允许代理向 Cisco Secure Workload 注册，并且还处理配置推送和软件升级通知。
- 代理 -> 思科 Cisco Secure Workload TCP/443 (TLS) 上的 SaaS 流数据（收集器）
流数据是提取的流元数据信息；数据将一次发送到一组 16 个 IP 地址。第二组 IP 地址用于备用设备。这大约是实际服务器流量的 1-5%。
- 代理 -> Cisco Secure Workload TCP/443 (TLS) (efe) 上的 SaaS 执行数据
执行数据通道是一个低容量控制通道，用于将策略推送到传感器，并收集执行统计信息。

传感器根据随代理安装的本地 CA 验证来自 Cisco Secure Workload 云控制、数据和执行服务器的 TLS 证书。由于不使用其他 CA，因此发送到代理的任何其他证书都将导致验证失败，并且代理将无法连接。这将导致代理无法注册、签入、发送流或接收执行策略。

配置代理通信的 IP 流量

大多数情况下，典型的配置是在代理（工作流程）和 Cisco Secure Workload TaaS 之间设置边界防火墙和可能的代理。



Note Cisco Secure Workload 会在载入期间收集网关/NAT IP 信息，并在创建租户时自动添加信息。如果您在门户中添加新的 IP 地址或更改 IP 地址，这些更改需要 Cisco Secure Workload 工作人员的审查和批准。

除了在 TaaS 门户网站上添加网关/NAT IP 地址外，可能还需要对网络进行更多改动，以允许流量出站且不进行修改：

- 在边界防火墙上允许通过 TLS/HTTPS 出站端口 443。
- 如果使用的是解密 Web 代理，请在 Web 代理上配置代理绕行和 SSL/TLS 绕行。
- 如果在数据中心使用透明 Web 代理，则必须路由特定的 SaaS IP 地址并配置绕行规则。传感器是无法执行自动 HTTPS 重定向的连接。

与代理通信的 IP 列表可在 TaaS 门户网站上查看。要添加到防火墙出站配置和代理绕行的 IP 标记为 collector-n、efe-n（仅在部署执行时）和 sensorVIP。通常需要添加 17 到 33 个 IP 用于代理通信，但也可能更多或更少，具体取决于您的 TaaS 配置。

SSL/TLS 连接故障排除

如上一部分所述，配置显式或透明 Web 代理，以绕过 SSL/TLS 解密进行代理通信非常重要。如果没有配置绕行，这些代理可能会尝试解密

SSL/TLS 流量，向代理发送自己的证书。由于代理只使用本地 CA 验证证书，因此这些代理证书会导致连接失败。

症状包括代理无法注册到集群、代理未签入、代理未发送流和/或代理未接收执行配置（如果已启用执行）。



Note 以下故障排除步骤假设使用了默认安装路径。Windows: C:\Program Files\Cisco Tetration Linux: /usr/local/tet。如果您将代理安装在其他位置，请在说明中替换该位置。

代理日志中会报告 SSL/TLS 连接问题。要验证日志中是否存在 SSL 错误，请针对观察到的相关问题运行以下命令。

注册、签入

Linux

```
grep "NSS error" /usr/local/tet/log/check_conf_update.log
```

Windows (PowerShell)

```
get-content "C:\Program Files\Cisco Tetration\logs\TetUpdate.exe.log" | select-  
->string -pattern "curl failed SSL peer certificate"
```

流

大多数 SSL/TLS 连接问题都发生在初始连接和代理注册期间。发送流需要在尝试连接前完成注册。这里出现的 SSL/TLS 错误是由于允许使用 sensorVIP IP 但不允许使用收集器 IP 所致。

Linux

```
grep "SSL connect error" /usr/local/tet/log/tet-sensor.log
```

Windows (PowerShell)

```
get-content "C:\Program Files\Cisco Tetration\logs\WindowsSensor*.log" | select-  
->string -pattern "Certificate verification error"
```

执行

Linux

```
grep "Unable to validate the signing cert" /usr/local/tet/log/tet-enforcer.log
```

Windows (PowerShell)

```
get-content "C:\Program Files\Cisco Tetration\logs\WindowsSensor*.log" | select-  
->string -pattern "Handshake failed"
```

如果在上述日志检查中发现 SSL 错误，可以使用以下命令验证发送给代理的证书。

显式代理 - 其中代理在 user.cfg 中配置

Linux

```
curl -v -x http://<proxy_address>:<port> https://<sensorVIP>:443
```

Windows (PowerShell)

```
cd "C:\Program Files\Cisco Tetration"
.\curl.exe -kv -x http://<proxy_address>:<port> https://<sensorVIP>:443
```

透明代理 - 无需 `user.cfg` 代理配置。它是一个代理，在从代理到互联网的所有 HTTP(S) 流量之间配置。

Linux

```
openssl s_client -connect <sensorVIP from TaaS Portal>:443 -CAfile /usr/local/tet/
→cert/ca.cert
```

Windows (PowerShell)

```
cd C:\Program Files\Cisco Tetration
.\openssl.exe s_client -connect <sensorVIP from TaaS Portal>:443 -CAfile cert\ca.cert
```

您正在 `openssl s_client` 响应中查找以下内容

```
Verify return code: 0 (ok)
```

如果出现错误，请检查证书。证书（链）示例应仅包括以下证书（CN IP 为示例）：

证书链

```
0 s:/C=US/ST=CA/L=San Jose/O=Cisco Systems, Inc./OU=Tetration, Insieme BU/CN=129.146.
→155.109
1:/C=US/ST=CA/L=San Jose/O=Cisco Systems, Inc./OU=Tetration Analytics/CN=Customer CA
```

如果您看到其他证书，则代理和 Cisco Secure Workload 之间可能存在 Web 解密代理。请联系您的安全或网络小组，确认是否已使用“为代理通信配置 IP 流量”部分中列出的 IP 配置了代理绕行。

Windows 2016 服务器上的 Windows 传感器安装脚本失败：可能出现的错误消息“The underlying connection was closed: An unexpected error occurred on a receive.”。可能的原因可能是 PowerShell 中设置的 SSL/TLS 版本。

要检查正在运行的 SSL/TLS 版本，请运行以下命令：

```
[Net.ServicePointManager]::SecurityProtocol
```

如果上述命令的输出为：

```
Ssl3, Tls
```

然后使用下面的命令更改允许的协议并重试安装：

```
[Net.ServicePointManager]::SecurityProtocol = [System.Net.SecurityProtocolType]'Ssl3,
→Tls,Tls11,Tls12'
```

代理操作

问：我已成功安装代理，但在“UI 传感器监控” (UI Sensor Monitoring) 页面上没有看到它。

答：代理需要先注册到在集群中运行的后端服务器，然后才能开始运行。如果 UI 页面上未显示代理，很可能是因为注册失败。我们可以从几个方面来了解注册失败的原因：

- 检查代理与后台服务器之间的连接是否正常

- 检查 curl 请求是否能正确发送到后端服务器
- 检查 HAProxy 访问和后端服务器日志，以查看服务器是否收到注册请求
- 检查日志文件中 curl 请求返回的错误消息

问：代理已安装，我可以在 UI 页面上找到。但是，“软件版本” (SW Ver) 列显示“正在初始化” (initializing)，而不是版本字符串。

答：在后端服务器安装并注册初始代理后，该代理还需要 30 分钟来报告其版本。

问：代理已正确升级，但经过很长时间（比如几个小时）后，“软件版本” (SW Ver) 字段仍显示旧版本。

答：代理成功升级后，会尝试发送 curl 请求以报告其当前运行的版本，并在同一请求中检查新版本。由于以下几个原因，请求可能无法到达后端：

- 请求超时，无法及时获得响应
- 网络存在问题，代理无法连接到后端服务器

问：我有一个在 RHEL/CentOS-6.x 上运行的代理，它运行正常。我正打算将操作系统升级到 RHEL/CentOS-7.x。升级后，代理还能工作吗？

答：目前，我们不支持升级操作系统的场景，尤其是升级主要版本。要使代理在操作系统升级后正常工作，请执行以下步骤：

- 卸载现有代理软件
- 清理所有文件，包括证书
- 转到 UI，删除代理条目
- 将操作系统升级到所需版本
- 在新操作系统上安装代理软件

问：我有一个在 RHEL/CentOS-6.x 上运行的代理，它运行正常。我打算为主机重命名。重命名/重启后，代理是否仍可工作？

答：代理身份是根据主机的唯一性（包括主机名和 BIOS-UUID）计算得出的。更改主机名会更改主机的标识。建议执行以下操作：

- 卸载现有代理软件
- 清理所有文件，包括证书
- 转到 UI，删除旧代理条目
- 重命名主机并重启
- 再次安装代理软件

问：在 Windows 主机上，添加/删除/修改规则导致防火墙偏离。如何查找规则？

答：在检测偏差时，代理会将最近 15 秒的防火墙事件记录到“C:\Windows\System32\config\systemprofile\AppData\Roaming\tet\firewall_events”。导致偏差的规则将在创建为 policy_dev_<policy id>_<timestamp>.txt 的最新文件中找到。

问：我已在 Windows 主机上成功安装代理。为什么我看不到传感器的任何报告流？

答：在 Windows 主机上收集流需要使用 Npcap。成功安装代理后 10 秒，它将安装 Npcap。如果传感器在几分钟内未报告流量，请检查代理和后端服务器是否已连接，以及 Npcap 是否已在 [Npcap 问题](#) 上正确安装。

问：我已在 Windows 主机 2008 R2 上成功安装代理。为什么 tetsensor 服务运行时系统时钟会漂移？

答：这是 Go 和 Windows 2008 R2 的已知问题。有关详细信息，请参阅 [Golang](#) 和 [Win2008 R2](#)。

作为 tetsensor 服务一部分运行的进程 tet-main.exe，是使用 Go 1.15 版本构建的。这就是 tetsensor 服务运行时系统时钟漂移的原因。

当 Windows 2008 R2 工作负载配置为使用外部 NTP 服务器或域控制器作为 NTP 服务器时，就会出现此问题。

可能的解决方法：

1. 定期强制 NTP 同步时钟：w32tm /resync /force

2. 手动禁用 tet-main.exe。

- 使用“管理员”权限运行 cmd.exe。
- 运行 regedit.exe
- 转到“HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\TetSensor”
- 双击“ImagePath”
- 编辑值，删除 tet-main.exe

在“C:\Program Files\Cisco Tetration\TetSenEngine.exe” TetSensor TetSen.exe “-f sensor_config” tet-main.exe ” ” TetUpdate.exe 之前

在“C:\Program Files\Cisco Tetration\TetSenEngine.exe” TetSensor TetSen.exe “-f sensor_config” TetUpdate.exe 之后

- 重启 tetsensor 服务



Note 每次升级代理后，请禁用 tet-main.exe。

3. 删除外部 NTP 服务器配置：

- 运行命令：w32tm /config /update /manualpeerlist: /syncfromflags:manual /reliable:yes
- 重启 Windows 时间服务 W32Time

代理故障排除工具

代理故障排除工具可让您在 Windows 环境中排除代理的常见问题。此工具是一个带有多个参数的 PowerShell 脚本，可让您对代理的不同方面进行故障排除。以下 PowerShell 参数可用于对代理的不同方面进行故障排除：

- `-agentHealth`：此选项可检查代理的运行状况并报告需要解决的任何问题。
- `-agentRegistration`：此选项允许您检查代理注册是否存在任何已知问题。
- `-agentUpgrade`：此选项允许您检查代理升级是否存在任何已知问题。
- `-enforcementHealth`：此选项可检查整体执行运行状况，并确保编程了最新的策略。
- `-collectLogs`：此选项可收集调试日志，从而对其进行分析以作进一步的故障排除。

要运行代理故障排除工具脚本，请按照以下步骤进行操作：

过程

步骤 1 以管理员身份打开 Windows (PowerShell)。

步骤 2 导航至 CSW 安装目录（此目录的默认位置为：“C:\Program Files\Cisco Tetration”）。

步骤 3 使用以下命令运行脚本：

```
.\AgentTroubleshooting.ps1
```

例如，要检查代理的运行状况，请使用 `-agentHealth` 参数运行脚本：

```
.\AgentTroubleshooting.ps1 -agentHealth
```

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。