



## Software Agents

---

A Secure Workload software agent is a lightweight piece of software that you install on your workloads. Its purpose is to:

- Collect host information such as network interfaces and active processes running in the system.
- Monitor and collect network flow information.
- (When enabled) Enforce security policies by setting firewall rules on the installed hosts.

Agents automatically update your Secure Workload inventory when interface addresses change.

You do not need to install agents on end-user (employee) computers.

- [Deploy Software Agents, on page 1](#)
- [Security Exclusions, on page 25](#)
- [Service Management of Agents, on page 26](#)
- [Policy Enforcement with Agents, on page 28](#)
- [Software Agent Config, on page 52](#)
- [View Detailed Agent Status in the Workload Profile, on page 63](#)
- [Rehoming of Agents, on page 65](#)
- [Host IP Address Change when Enforcement is Enabled, on page 68](#)
- [Upgrading Software Agents, on page 69](#)
- [Removing Software Agents, on page 72](#)
- [Data collected and exported by workload agents, on page 74](#)
- [Enforcement Alerts, on page 76](#)
- [Sensor Alerts, on page 82](#)

## Deploy Software Agents



---

**Note** Installer scripts downloaded from LDAP or AD accounts with automatic role mapping fail once you are logged out. To give the installer scripts uninterrupted access to the cluster, enable Use Local Authentication.

---

On deployment, the agent is assigned a unique identity by the Secure Workload cluster, based on a set of parameters specific to the host where the agent is running. If the host name and the BIOS UUID are a part of the set of parameters, you may encounter the following issues:

1. Registration failure when cloning a virtual machine and retaining the BIOS UUID and host name, and when instant cloning a VDI. The registration failure happens because Secure Workload already has a registered software agent using the same parameters set. You can delete the registered agent using OpenAPI. In some cases, a duplicate BIOS UUID configured during startup is changed by VMware after a certain period of time. Agent registration recovers once the Cisco Secure Workload services are restarted.
2. A new identity is generated for the agent if the host name is changed and the host rebooted. The redundant or the old agent is marked as inactive after a certain period of time. For more information, see Frequently Asked Questions section.

## Supported Platforms and Requirements

For information on supported platforms and additional requirements for software agents, see:

- The release notes for your release, see [Release Notes](#).
- The agent install wizard in the Secure Workload web portal: In the navigation menu, click **Manage > Workloads > Agents**, then click the **Installer** tab. Choose an installation method, a platform, and if applicable, an agent type to see supported platform versions.
- [Support Matrix](#) for additional dependencies.
- The following sections for details on additional requirements for each platform and agent type.

## Installing Linux Agents for Deep Visibility and Enforcement

### Requirements and Prerequisites to Install Linux Agents

- See [Supported Platforms and Requirements](#).
- Root privileges to install and execute the services.
- 1-GB storage space for agent and log file.
- Security exclusions are configured on the security applications that are monitoring the host to prevent these applications from blocking agent installation or agent activity. For more information, see [Security Exclusions](#).
- A special user, **tet-sensor**, is created in the host where the agent is installed. If PAM or SELinux is configured on the host, tet-sensor user must be granted appropriate privileges for executing the tet-sensor process and making connections to collectors. If an alternative install directory is provided and SELinux is configured, ensure that execution is allowed for that location.
- You must be able to use the unzip command, if the agent is installed using the AutoInstall (installer script) method.

### Supported Methods to Install Linux Agents

Methods to install a Linux agent for deep visibility and enforcement:

- [Install Linux Agent Using the Agent Script Installer Method, on page 3](#)
- [Install Linux Agent using the Agent Image Installer Method, on page 5](#)

## Install Linux Agent Using the Agent Script Installer Method

We recommend the installer script method to deploy Linux agents for deep visibility and enforcement.



- 
- Note**
- The installed Linux agent supports both deep visibility and enforcement.
  - By default, enforcement is disabled. To enable enforcement, see [Create an Agent Configuration Profile](#).
- 

To install a Linux agent using the script installer method:

### Step 1

Navigate to Agent Installation methods:

- If you are a first-time user, launch the **Quick Start Wizard** and click **Install Agents**.
- In the left pane, click **Manage > Agents**, and select the **Installer** tab.

### Step 2

Click **Agent Script Installer**.

### Step 3

From the **Select Platform** drop-down list, choose **Linux**.

To view the supported Linux platforms, click **Show Supported Platforms**.

### Step 4

Choose the tenant to install the agents.

**Note** Secure Workload SaaS clusters do not require selecting a tenant.

### Step 5

If you want to assign labels to the workload, choose the label keys and enter label values.

When the installed agent reports IP addresses on the host, the installer CMDB labels selected here, along with other uploaded CMDB labels that have been assigned to IPs reported by this host, would be automatically assigned to the new IP address. If there are conflicts between uploaded CMDB labels and installer CMDB labels:

- Labels assigned to an exact IP address take precedence over labels assigned to the subnet.
- Existing labels assigned to an exact IP address take precedence over installer CMDB labels.

### Step 6

If an HTTP proxy is required to communicate with Secure Workload, choose **Yes**, and then enter a valid proxy URL.

### Step 7

In the **Installer expiration** section, select an option:

- No expiration: The installer script can be used multiple times.
- One time: The installer script can be used only once.
- Time bound: You can set the number of days for which the installer script can be used.
- Number of deployments: You can set the number of times the installer script can be used.

### Step 8

Click **Download** and save the file to the local disk.

### Step 9

Copy the installer shell script on Linux hosts and run the following command to grant execute permission to the script:

```
chmod u+x tetration_installer_default_sensor_linux.sh
```

**Note** The script name may differ depending on the selected agent type and scope.

**Step 10** To install the agent, run the following command with root privileges:

```
./tetration_installer_default_sensor_linux.sh
```

**Note** If an agent is already installed on the tenant, you cannot proceed with the installation.

---

We recommend running the precheck, as specified in the script usage details.

**Linux installer script usage details:**

```
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=<logbasedir>]
  [--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 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 uses the native .deb package, and new installations and reinstallations switch to this package type. Upgrades from previous versions continue with the .rpm package.
- Ubuntu .deb package is installed under /opt/cisco/tetration.
- There is no relocation support for the .deb package and so the `--basedir` option is not supported for Ubuntu.

## Install Linux Agent using the Agent Image Installer Method

We recommend the automated installer script method for installing Linux agents. Use the image installer method if you have a specific reason for using this manual method.

**Prerequisite:**

Configure the `ACTIVATION_KEY` and `HTTPS_PROXY` in the `user.cfg` file for SaaS clusters and when you are installing the agent on a non-default tenant of on-premises clusters with multiple tenants. For more information, see [\(Manual Installations Only\) Update the User Configuration File](#).

To install a Linux agent using the agent image method:

**Step 1**

Navigate to Agent Installation Methods:

- If you are a first-time user, launch the Quick Start wizard and click **Install Agents**.
- In the navigation pane, click **Manage > Agents**, and select the **Installer** tab.

**Step 2**

Click **Agent Image Installer**.

**Step 3**

In the **Platform** field, enter Linux.

**Step 4**

Enter the required agent type and the version of the agent, and then from the results, download the required version of the agent.

**Step 5**

Copy the RPM package to all the Linux hosts for deployment.

**Note**

If the agent is already installed on the host, do not reinstall the agent. To upgrade the agent, see [Upgrading Software Agents](#) section.

**Step 6**

Based on your platform, run the RPM commands with root privileges.

- For RHEL/CentOS/Oracle platforms, run command: `rpm -ivh <rpm_filename>`
- For Ubuntu platform:
  - To retrieve the dependency list and ensure all dependencies are met, run the command: `rpm -qpR <rpm_filename>`
  - Install the agent with “`--nodeps`” option by running the command: `rpm -ivh \\\--nodeps <rpm filename>`

## Verify Linux Agent Installation

---

Run command `sudo rpm -q tet-sensor` `sudo rpm -q tet-sensor`.

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

Verify that you have a single entry as the output, which confirms that a Linux agent is installed on the host.

Sample output: `tet-sensor-3.1.1.50-1.el6.x86_64`

The specific output may differ depending on the platform and architecture.

---

## Installing Windows Agents for Deep Visibility and Enforcement

### Requirements and Prerequisites for Installing Windows Agent

- For more information, see the Supported Platforms and Requirements.
- Administrator privileges are required for install and service execution.
- Npcap must be installed on workloads running Windows 2008 R2 or when the installed agent version is prior to version 3.8. If the Npcap driver is not already installed, the recommended Npcap version will be installed silently by the agent after the service starts.

For more information, see the Npcap version information.

- Storage requirement for agent and log files: 1 GB.
- Required Windows services: If your Windows hosts have been security that is hardened or have deviated from the default configuration as shipped from Microsoft, you may have some Windows services disabled that are required for successful agent installation.

For more information, see the Required Windows Services section.

- Prevent other security applications from blocking agent installation or agent activity by configuring security exclusions on the security applications that are monitoring the host. See

For more information, see the Security Exclusions section.

### Supported Methods to Install Windows Agents

There are two methods to install an agent on Windows platforms for deep visibility or enforcement:

- [Install Windows Agent using the Agent Script Installer Method, on page 6](#)
- [Install Windows Agent using the Agent Image Installer Method, on page 8](#)

You can also install using a golden image. See [Deploying Agents on a VDI Instance or VM Template \(Windows\)](#).

#### Install Windows Agent using the Agent Script Installer Method

The installer script is the recommended method to deploy agents on Windows platforms for deep visibility or enforcement.

**Note**

- The installed Windows agent supports both deep visibility and enforcement.
- By default, enforcement is disabled. To enable enforcement, see [Create an Agent Configuration Profile, on page 54](#).

To install a Windows agent using the script installer method:

**Step 1**

To navigate to the agent installation methods:

- For a first-time user, launch the Quick Start wizard and click **Install Agents**.
- In the left pane, click **Manage > Agents**, and select the **Installer** tab.

**Step 2**

Click **Agent Script Installer**.

**Step 3**

From the **Select Platform** drop-down menu, choose **Windows**.

To view the supported Windows platforms, click **Show Supported Platforms**.

**Step 4**

Choose the tenant to install the agents.

**Note**

Selecting a tenant is not required in Secure Workload SaaS clusters.

**Step 5**

(Optional) To assign labels to the workload, choose the label keys and enter label values.

When the installed agent reports IP addresses on the host, the installer CMDB labels selected here, along with other uploaded CMDB labels that have been assigned to IPs reported by this host, would be assigned to the new IP address automatically. If there are conflicts between uploaded CMDB labels and installer CMDB labels:

- Labels assigned to exact IP address take precedence over labels assigned to the subnet.
- Existing labels assigned to exact IP address take precedence over installer CMDB labels.

**Step 6**

If HTTP proxy is required to communicate with Secure Workload, choose **Yes**, and then enter a valid proxy URL.

**Step 7**

Under the **Installer expiration** section, select one from the available options:

- No expiration: The installer script can be used multiple times.
- One time: The installer script can be used only once.
- Time bounded: You can set the number of days the installer script can be used.
- Number of deployments: You can set the number of times the installer script can be used.

**Step 8**

Click **Download** and save the file to local disk.

**Step 9**

Copy the installer PowerShell script to all the Windows hosts for deployment and run the script with administrative privileges.

**Note**

- Depending on the system settings, the command `Unblock-File` may need to be run before other commands.

We recommend running the pre-check, as specified in the script usage details.

### Windows installer script usage details:

```
# powershell -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"
```

### Install Windows Agent using the Agent Image Installer Method

We recommend that the installer script method (automated) is used to install the Windows agents, unless you have a specific reason to install the agent manually using the Agent Image method.




---

**Note** Do not manually deploy an older MSI agent version when an existing agent is already running on the host.

---



Site-related files in the package:

- **ca.cert**—Mandatory—CA certificate for sensor communications.
- **enforcer.cfg**—Mandatory only when installing enforcement sensor—Contains configuration of enforcement endpoints.
- **sensor\_config**—Mandatory—Configuration for deep visibility sensor.
- **sensor\_type**—Type of the sensor (enforcement or deep visibility).
- **site.cfg**—Mandatory—Global site endpoint configuration.
- **user.cfg**—Mandatory for SaaS—Sensor activation key and proxy configuration.

#### Prerequisite:

For SaaS clusters and when you are not installing under the default tenant on on-premises clusters with multiple tenants, you must configure the `ACTIVATION_KEY` and `HTTPS_PROXY` in the `user.cfg` file. For more information, see [\(Manual Installations Only\) Update the User Configuration File](#).

To install a Windows agent using the agent image method:

- 
- Step 1** To navigate to the agent installation methods:
- For a first-time user, launch the Quick Start wizard and click **Install Agents**.
  - In the left pane, click **Manage > Agents**, and select the **Installer** tab.
- Step 2** Click **Agent Image Installer**.
- Step 3** In the **Platform** field, enter Windows.
- Step 4** Enter the required agent type and the version of the agent, and then from the results, download the required version of the agent.
- Step 5** Copy the `tet-win-sensor<version>.win64-<clustertype>.zip` file to all the Windows hosts for deployment.
- Step 6** With administrative privileges, extract the ZIP file.
- Step 7** In the extracted folder, run the following command to install the agent: `msiexec.exe /i TetrationsAgentInstaller.msi`  
Additionally, the following options are available for MSI installer.

**Table 1: Available Options for MSI Installer**

Options	Description
<code>agenttype=&lt;AgentType&gt;</code>	<i>AgentType</i> should either be <i>sensor</i> or <i>enforcer</i> , depending on whether enforcement is required. By default, content of <code>sensor_type</code> file is checked by the installer in same folder and overwrites the passed parameter. However, if agent is installed in <i>/quiet</i> mode, this option is required.
<code>overwrittenpcap=yes</code>	For Windows 2008 R2, by default, the agent will not attempt to upgrade Npcap if Npcap already exists. Pass this parameter to upgrade the existing Npcap. If this option is used, subsequent agent auto-upgrades will also upgrade Npcap to newer supported versions.

Options	Description
nostart=yes	You must pass this parameter when installing the agent on a golden image in a VDI environment or VM template to prevent agent services (tetsensor and tetenforcer) from starting automatically. On VDI/VM instances created from the golden image with a different host name, these services will start automatically, as expected.
installfolder=<FullPathCustomFolder>	Use the parameter at the end of the install command to install agent in a custom folder.
serviceuser=<Service UserName>	Use the parameter at the end of the install command to configure service user. Default service user is "LocalSystem".  For local user, serviceuser=.\<Service UserName>  For domain user, serviceuser=<domain_name>\<samaccount name>  Service user must have local administrative privileges.
servicepassword=<Service UserPassword>	Use the parameter at the end of the install command to configure password for the service user. The password must be in plain-text format.
proxy="<proxy_address>"	Use the parameter to set HTTPS proxy to access the Secure Workload cluster.
activationkey=<activation Key>	Use the parameter to specify tenant if agent is not installed under the default tenant.

**Note**

- If activation key and proxy options are used during manual installation, you do not need to manually configure *user.cfg*.
- If Npcap is not installed on the host, the Secure Workload service, tetsensor installs Npcap automatically.
- If the agent is already installed on the host, do not reinstall the agent. To upgrade the agent, see Upgrading Software Agents section.

## Verify Windows Agent Installation

**Step 1** Verify that the folder `C:\Program Files\Cisco Tetration` (or the custom folder) exists.

**Step 2** Verify that the service—*TetSensor*, for deep visibility, exists and is in the running state. Run command `cmd.exe` with administrative privileges.

Run command `sc query tetsensor`

Check state **Running**

Run command `sc qc tetsensor`

Check DISPLAY-NAME **Cisco Secure Workload Deep Visibility**

OR

Run command `services.msc`

Find name **Cisco Secure Workload Deep Visibility**

Check status **Running**

### Step 3

Verify that the service—*TetEnforcer*, for enforcement, exist and is in the running state.

Run command `cmd.exe` with **Admin** privileges

Run command `sc query tetenforcer`

Check state **Running**

Run command `sc qc tetenforcer`

Check DISPLAY-NAME **Cisco Secure Workload Enforcement**

OR

Run command `services.msc`

Find name **Cisco Secure Workload Enforcement**

Check status **Running**

## Verify Windows Agent in the Configured Service User Context

1. Verify that the service TetSensor (for deep visibility) and TetEnforcer (for enforcement) running in the config- ured service user context. TetSensor and TetEnforcer run in the same service user context.

Run command `cmd.exe` with **Admin** privileges

Run command `sc qc tetsensor`

Check SERVICE\_START\_NAME <configured service user>

Run command `sc qc tetenforcer`

Check SERVICE\_START\_NAME <configured service user>

OR

Run command `services.msc`

Find name **Cisco Secure Workload Deep Visibility**

Check **Log On As** for the <configured service user>

Find name **Cisco Secure Workload Enforcement**

Check **Log On As** for the <configured service user>

OR

Run command `tasklist /v | find /i "tet"`

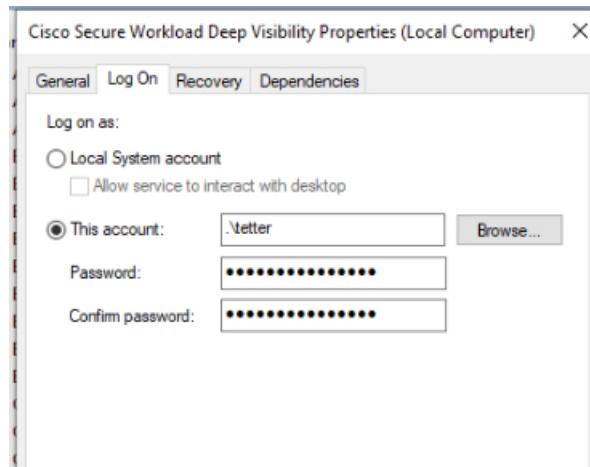
Check the user context for the running processes (5th column)

## Modify Service Account

To modify the service account after installing Windows Agents, use one of the following methods to modify the existing Deep Visibility and Enforcement services.

- Use **services.msc**.

**Figure 1: Modify Service Account based on services.msc Account**



- Use any third party application to configure the services.
- Use the following commands:
  1. Run cmd as an administrator.
  2. Modify the services using service account name by running the following commands:
    - a. `sc config tetsensor obj= <service user name> password= <password>`
    - b. `sc config tetenforcer obj= <service user name> password= <password>`
  3. Verify service configurations by running the following commands:
    - a. `sc qc tetsensor`
    - b. `sc qc tetenforcer`
  4. Restart the tetsensor and tetenforcer services by running the following commands:
    - a. `sc stop tetsensor / tetenforcer`
    - b. `sc start tetsensor / tetenforcer`

## Deploying Agents on a VDI Instance or VM Template (Windows)

By default, agent services start automatically after agents are installed. When installing on a golden image, you must use installer flags to prevent these services from starting. When instances are cloned from the golden image, agent services will start automatically as expected.

Similarly, NPCAP is normally installed automatically after agent installation if it is not already present. NPCAP is not automatically installed on golden images, but will be automatically installed if needed on VM instances cloned from a golden image. For more information, see [Windows Agent Installer and Npcap](#).

### Install the agent on a golden image in a VDI environment or VM template

---

**Step 1** Install the agent on a golden image in a VDI environment or VM template using an MSI installer or PowerShell installer script:

Using MSI installer with **nostart=yes**

- For more information, see [Install Windows Agent using the Agent Image Installer Method, on page 8](#).
- `msiexec.exe /<MSI installer> nostart="yes" /quiet /norestart /! *v <installer_log_file>` OR

OR

Using PowerShell installer with the **-goldenImage** flag.

- For more information, see [Install Windows Agent using the Agent Script Installer Method, on page 6](#).

**Step 2** Verify that the folder `C:\Program Files\Cisco Tetration` (or the custom folder) exists.

**Step 3** Verify that the service TetSensor (for deep visibility) exists and is stopped:

Run command `cmd.exe` with **Admin** privileges

Run command `sc query tetsensor`

Check STATE **Stopped**

**Step 4** Verify that the service TetEnforcer (for enforcement) exists and is stopped:

Run command `sc query tetenforcer`

Check STATE Stopped

**Step 5** The VM template is now configured.

**Step 6** Shut down the VM template.

---

### Create a new VDI instance VM

---

**Step 1** Create a new VDI instance VM by cloning the VM template.

**Step 2** Reboot the VDI instance VM.

**Step 3** After rebooting the VDI instance VM, verify that the services – TetSensor (for deep visibility) and TetEnforcer (for enforcement) – are running in the configured service context. See [Verify Windows Agent Installation](#).

**Step 4** On the VDI instance VM, verify that the NPCAP driver is installed and running:

Run command `cmd.exe` with Admin privileges

Run command `sc query npcap`

Check STATE **Running**

**Step 5** On the VDI instance VM, verify that the agent is registered using a valid `sensor_id`:

- Check the `sensor_id` file in installation folder.
- If the `sensor_id` starts with “uuid”, it is not a valid `sensor_id`.
- If the agent fails to register but the Secure Workload web interface shows that the agent is registered:
- Delete the agent using OpenAPI. For more information, see [Deploy Software Agents](#).

**Note**

- Do not change the host name of the golden image or VM template.
- If the golden image or VM template is rebooted after installing the agent, Secure Workload services will start running after reboot.
- If the VDI instance VM fails to report network flows, see the VDI Instance VM in Network Flows section.

---

## Windows Agent Installer and Npcap

1. For supported Npcap versions, see the Support Matrix at <https://www.cisco.com/go/secure-workload/requirements/agents>.

2. Installation:

If Npcap is not installed, the agent will install the supported version ten seconds after the service starts. If User has Npcap installed but is older than supported version, Npcap will not be upgraded. Upgrade/uninstall Npcap yourself, run the agent installer with option **overwrittenpcap=yes**, or run installer script with **-npcap** to get the supported Npcap version. If Npcap driver is in use by any application, the agent will try to upgrade Npcap at a later time.

3. Upgrade:

If Npcap is installed by Windows Agent and the version is older than the supported version, Npcap will be upgraded to the supported version ten seconds after the service starts. If Npcap driver is in use by any application, the agent will try to upgrade Npcap at a later time. If Npcap is not installed by Windows Agent, Npcap will not be upgraded.

4. Uninstall:

If Npcap is installed by Windows Agent, it will uninstall Npcap. If Npcap is installed by user, but upgraded by the agent installer with **overwrittenpcap=yes**, it will not be uninstalled. If Npcap driver is in use by any application, the agent will not uninstall Npcap.

# Installing AIX Agents for Deep Visibility and Enforcement



**Note** Process tree, Package (CVE), and Forensic Event reporting features are not available on AIX. Additionally, some aspects of those features may not be available on specific minor releases of otherwise supported platforms due to OS limitations.

## Requirements and Prerequisites for Installing AIX Agents

- See [Supported Platforms and Requirements](#).
- Additional requirements for deep visibility:
  - Root privileges are required to install and execute the services.
  - Storage requirement for agent and log files: 500 MB.
  - Prevent other security applications from blocking agent installation or agent activity by configuring security exclusions on any security applications that are monitoring the host. See [Security Exclusions](#).
  - AIX only supports flow capture of 20 net devices (6 net devices if version is AIX 7.1 TL3 SP4 or earlier). The deep visibility agent captures from a maximum of 16 network devices, leaving the other 4 capture sessions available for exclusive generic system usage (For example, tcpdump).
  - The deep visibility agent does the following to ensure this behaviour:
    - The agent creates 16 bpf device nodes under the agents directory (/opt/cisco/tetration/chroot/dev/bpf0 - /opt/cisco/tetration/chroot/dev/bpf15)
    - tcpdump and other system tools using bpf will scan through the system device nodes (/dev/bpf0-/dev/bpf19) until they find an unused node (!EBUSY)
    - The agent created bpf nodes and system bpf nodes will share the same major/minor, with each major/minor only be opened by one instance (either tcpdump or agent)
    - The agent will not access the system device nodes, and not create them as tcpdump does (tcpdump-D will create /dev/bpf0. . . /dev/bpf19 if they do not exist).
  - Running ipttrace on system will prevent in certain scenarios flow capture from tcpdump and deep visibility agent. This is a known design issue and needs to be checked with IBM.
    - To check if this scenarios exists before installing the agent, run tcpdump. If error message is like **tcpdump: BIOCSETIF: en0: File exists** ipttrace is blocking flow capture. Stopping ipttrace will resolve the issue.
  - Not every deep visibility functionality is supported on AIX. Package and process accounting are among the ones not supported.
- Additional requirements for policy enforcement:
  - If IP Security Filter is enabled (i.e. smitty ipsec4), agent installation fails in pre-check. We recommend to disable IP Security Filter before installing agent.

- When IP security is enabled, while Secure Workload enforcer agent is running, it will be reported as an error and enforcement agent will stop enforcing. To disable IP Security Filter when enforcement agent is active, contact a customer service representative.

## Install AIX Agent using the Agent Script Installer Method

Deep visibility or enforcement AIX agents can only be installed using the Agent Script Installation method.



### Note

- The installed AIX agent supports both deep visibility and enforcement.
- By default, enforcement is disabled. To enable enforcement, see [Create an Agent Configuration Profile, on page 54](#).

To install an AIX agent:

### Step 1

To navigate to the agent installation methods:

- For a first-time user, launch the Quick Start wizard and click **Install Agents**.
- In the left pane, click **Manage > Agents**, and select the **Installer** tab.

### Step 2

Click **Agent Script Installer**.

### Step 3

From the **Select Platform** drop-down menu, choose **AIX**.

To view the supported AIX platforms, click **Show Supported Platforms**.

### Step 4

Choose the tenant to install the agents.

**Note** Selecting a tenant is not required in Secure Workload SaaS clusters.

### Step 5

(Optional) To assign labels to the workload, choose the label keys and enter label values.

When the installed agent reports IP addresses on the host, the installer CMDB labels selected here, along with other uploaded CMDB labels that have been assigned to IPs reported by this host, would be assigned to the new IP address automatically. If there are conflicts between uploaded CMDB labels and installer CMDB labels:

- Labels assigned to exact IP address take precedence over labels assigned to the subnet.
- Existing labels assigned to exact IP address take precedence over installer CMDB labels.

### Step 6

If HTTP proxy is required to communicate with Secure Workload, choose **Yes**, and then enter a valid proxy URL.

### Step 7

Under the **Installer expiration** section, select one from the available options:

- No expiration: The installer script can be used multiple times.
- One time: The installer script can be used only once.
- Time bounded: You can set the number of days the installer script can be used.
- Number of deployments: You can set the number of times the installer script can be used.



- Step 8** Click **Download** and save the file to local disk.
- Step 9** Copy the installer shell script to all the AIX hosts for deployment.
- Step 10** To grant execute permission to the script, run command: `chmod u+x tetration_installer_default_sensor_aix.sh`
- Note** The script name may differ depending on agent type and scope.
- Step 11** To install the agent, run the following command with root privileges: `./tetration_installer_default_sensor_aix.sh`
- Note** If an agent is already installed on the host, you cannot proceed with the installation.

We recommend running the pre-check, as specified in the script usage details.

#### AIX installer script usage details:

```
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

```

## Verify AIX Agent Installation

---

Run command `lslpp -c -l tet-sensor.rte`, confirm that there is one entry as follows.

**Note** The specific output may differ depending on the version

```
$ sudo lslpp -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

---

## Installing Kubernetes or OpenShift Agents for Deep Visibility and Enforcement

### Requirements and Prerequisites

Operating system support information is available at [Agent OS support matrix](#).

#### Requirements

- The install script requires Kubernetes or OpenShift administrator credentials to start privileged agent pods on the cluster nodes.
- Secure Workload entities are created in the **tetration** namespace.
- The node or pod security policies must permit privileged mode pods.
- busybox:1.33 images must either be preinstalled or downloadable from Docker Hub.
- For containerd run time, if the `config_path` is not set, modify your `config.toml` (default location: `/etc/containerd/config.toml`) as follows:

```

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

```

Restart the containerd daemon.

- In order to run on Kubernetes or OpenShift control plane nodes, the `-toleration` flag can be used to pass in a toleration for the Secure Workload pods. This usually is the NoSchedule toleration that normally prevents pods from running on control plane nodes.
- For Windows worker nodes:
  - Supported Windows worker node container runtime: ContainerD.
  - ContainerD config: Configure the following containerd change.

```

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

```

Remove configurations under **registry.mirrors**. The default configuration file location is `C:\Program Files\containerd\config.toml`.

Restart the containerd daemon after the configuration changes.

- The image **mcr.microsoft.com/oss/kubernetes/windows-host-process-containers-base-image:v1.0.0** must either be preinstalled or downloadable on the Windows worker node.
- The existing Kubernetes agent which is upgrading to the newer version includes the Windows DaemonSet agent automatically. However, the previous script does not uninstall the Windows DaemonSet agent. Download the latest installer script to uninstall the Windows DaemonSet agent.
- Supported on:
  - Microsoft Windows Server 2022
  - Windows Server 2019
  - Kubernetes 1.27 and later

### Requirements for Policy Enforcement

IPVS-based kube-proxy mode is not supported for OpenShift.

These agents should be configured with the Preserve Rules option that is enabled. See [Create an Agent Configuration Profile](#).

For enforcement to function properly, any installed CNI plug-in must:

- Provide flat address space (IP network) between all nodes and pods. Network plug-ins which masquerade the source pod IP for intracluster communication are not supported.
- Not interfere with Linux iptables rules or marks that are used by the Secure Workload Enforcement Agent (mark bits 21 and 20 are used to allow and deny traffic for NodePort services)

The following CNI plug-ins are tested to meet the requirements above:

- Calico (3.13) with the following Felix configurations: (*ChainInsertMode: Append, IptablesRefreshInterval: 0*) or (*ChainInsertMode: Insert, IptablesFilterAllowAction: Return, IptablesMangleAllowAction: Return, IptablesRefreshInterval: 0*). All other options use their default values.

See the Felix configuration reference for more information on setting these options.

## Install Kubernetes or OpenShift Agent using the Agent Script Installer Method




---

**Note** The agent script installer method automatically installs agents on nodes included later.

---

**Step 1** To navigate to the agent installation methods:

- For a first-time user, launch the Quick Start wizard and click **Install Agents**.
- In the left pane, click **Manage > Agents**, and select the **Installer** tab.

**Step 2** Click **Agent Script Installer**.

**Step 3** From the **Select Platform** drop-down menu, choose **Kubernetes**.

To view the supported Kubernetes or OpenShift platforms, click **Show Supported Platforms**.

**Step 4** Choose the tenant to install the agents.

**Note** Selecting a tenant is not required in Secure Workload SaaS clusters.

**Step 5** If HTTP proxy is required to communicate with Secure Workload, choose **Yes**, and then enter a valid proxy URL.

**Step 6** Click **Download** and save the file to local disk.

**Step 7** Run the installer script on a Linux machine which has access to the Kubernetes API server and a kubectl configuration file with administrative privileges as the default context/cluster/user.

The installer will attempt to read the file from its default location (`~/.kube/config`). However, it be specified explicitly with the `--kubeconfig` command line option.

---

The installation script will provide instructions to verify the Secure Workload Agent Daemonset and Pods that were installed.




---

**Note** The HTTP Proxy configured on the agent installer page prior to download only controls how Secure Workload agents connect to the Secure Workload cluster. This setting does not affect how Docker images are fetched by Kubernetes or OpenShift nodes, since the container runtime on those nodes uses its own proxy configuration. If the Docker images are unable to be pulled from the Secure Workload cluster, debugging the container runtime's image pulling process will be necessary and adding a suitable HTTP proxy might be necessary.

---

## Installing Solaris Agents for Deep Visibility

### Requirements and Prerequisites for Installing Solaris Agents

- See [Supported Platforms and Requirements](#).

- Root privileges are required to install and execute the services.
- Storage requirement for agent and log files: 1 GB.
- Prevent other security applications from blocking agent installation or agent activity by configuring security exclusions on the security applications that are monitoring the host. See [Security Exclusions](#).

## Install Solaris Agent using the Agent Script Installer Method

The installed Solaris agent supports both deep visibility and process/ package visibility.

---

**Step 1** To navigate to the agent installation methods:

- For a first-time user, launch the Quick Start wizard and click **Install Agents**.
- In the left pane, click **Manage > Agents**, and select the **Installer** tab.

**Step 2** Click **Agent Script Installer**.

**Step 3** From the **Select Platform** drop-down menu, choose **Solaris**.

To view the supported Solaris platforms, click **Show Supported Platforms**.

**Step 4** Choose the tenant to install the agents.

**Note** Selecting a tenant is not required in Secure Workload SaaS clusters.

**Step 5** (Optional) To assign labels to the workload, choose the label keys and enter label values.

When the installed agent reports IP addresses on the host, the installer CMDB labels selected here, along with other uploaded CMDB labels that have been assigned to IPs reported by this host, would be assigned to the new IP address automatically. If there are conflicts between uploaded CMDB labels and installer CMDB labels:

- Labels assigned to exact IP address take precedence over labels assigned to the subnet.
- Existing labels assigned to exact IP address take precedence over installer CMDB labels.

**Step 6** If HTTP proxy is required to communicate with Secure Workload, choose **Yes**, and then enter a valid proxy URL.

**Step 7** Under the **Installer expiration** section, select one from the available options:

- No expiration: The installer script can be used multiple times.
- One time: The installer script can be used only once.
- Time bounded: You can set the number of days the installer script can be used.
- Number of deployments: You can set the number of times the installer script can be used.

**Step 8** Click **Download** and save the file to local disk.

**Step 9** Copy the installer shell script on Solaris hosts and run the following command to grant execute permission to the script:

```
chmod u+x tetratation_installer_default_sensor_solaris.sh
```

**Note** The script name may differ depending on the selected agent type and scope.

**Step 10** To install the agent, run the following command with root privileges:

```
./tetration_installer_default_sensor_solaris.sh
```

**Note** If an agent is already installed on the tenant, you cannot proceed with the installation.

We recommend running the pre-check, as specified in the script usage details.

#### Solaris installer script usage details:

```
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
```

## Verify Solaris Agent Installation

**Step 1** Run command: `sudo pkg list tet-sensor`

**Step 2** Verify that you have a single entry as the output, which confirms that a Solaris agent is installed on the host. Sample output:

NAME (PUBLISHER)	VERSION	IFO
tet-sensor (cisco)	3.8.1.1	i--

**Note** The specific output may differ depending on the platform and architecture.

## (Manual Installations Only) Update the User Configuration File

The following procedure is required only for installations involving *all* of the following:

- Secure Workload SaaS, or on-premises clusters with multiple tenants (on-premises clusters that use only the default tenant do NOT need this procedure)
- Manual installation
- Linux or Windows platform

In order for agents to register to the Secure Workload cluster, they require a cluster activation key. Additionally, if agents require an HTTPS proxy to reach the cluster, you must specify the proxy.



**Note** In Windows Environment, there is no need to configure `user.cfg` manually if `activationkey` and `proxy` options are used during manual installation.

Before installation, configure the required variables in the user configuration file:

**Step 1** Retrieve your activation key: Go to **Manage > Agents**, click the “Installer” tab, click “Manual Install using classic packaged installers”, then click “Agent Activation Key”.

**Step 2** Open the `user.cfg` file in the Secure Workload Agent installation folder for editing. (Example: `/usr/local/tet` on Linux or `C:\Program Files\Cisco Tetration` on Windows). The file contains a list of variables in the form of “key=value”, one on each line.

**Step 3** Add the activation key to the **ACTIVATION\_KEY** variable. Example:  
`ACTIVATION_KEY=7752163c635ef62e6568e9e852d07bd21bfd60d0`

**Step 4** If the agent requires an HTTPS proxy, add the **http** protocol proxy server and port using the **HTTPS\_PROXY** variable. Example: `HTTPS_PROXY=http://proxy.my-company.com:80`

## Other Agent-Like Tools

### AnyConnect agents

Platforms supported by Cisco AnyConnect Secure Mobility agent with Network Visibility Module (NVM). No additional Secure Workload agent is required. AnyConnect connector registers these agents and exports flow observations, inventories, and labels to Secure Workload. For more information, see [AnyConnect Connector](#).

For Windows, Mac, or Linux platforms, see [Cisco AnyConnect Secure Mobility Client Data Sheet](#).

### ISE agents

Endpoints registered with Cisco Identity Services Engine (ISE). No Secure Workload agent on the endpoint is required. ISE connector collects metadata about endpoints from ISE through pxGrid service on ISE appliance. It registers the endpoints as ISE agents on Secure Workload and pushes labels for the inventories on these endpoints. For more information, see [ISE Connector](#).

### SPAN agents

SPAN agents work with the ERSPAN connector. For information, see [ERSPAN Connector](#).

### Integration with third-party and additional Cisco products

- Integrations using external orchestrators configured in Secure Workload.  
See [External Orchestrators](#).
- Integrations using connectors configured in Secure Workload.  
See [What are Connectors](#).

## Connectivity Information

In general, when the agent is installed onto the workload, it starts making several network connections to the back-end services hosted on the Secure Workload cluster. Depending on the agent type and its functionality, the number of connections look different.

The following table captures various permanent connections that are made by various agent types.

**Table 2: Agent Connectivity**

Agent type	Config server	Collectors	Enforcement backend
visibility (on-premises)	CFG-SERVER-IP:443	COLLECTOR-IP:5640	N/A
visibility (SaaS)	CFG-SERVER-IP:443	COLLECTOR-IP:443	N/A
enforcement (on-premises)	CFG-SERVER-IP:443	COLLECTOR-IP:5640	ENFORCER-IP:5660
enforcement (SaaS)	CFG-SERVER-IP:443	COLLECTOR-IP:443	ENFORCER-IP:443
docker images	CFG-SERVER-IP:443	N/A	N/A



Legends:

- CFG-SERVER-IP represents the IP address of the config server.
- COLLECTOR-IP represents the IP address of the collector. Deep visibility and enforcement agents connect to all available collectors.
- ENFORCER-IP represents the IP address of the enforcement endpoint. The enforcement agent connects to only one of the available endpoints.
- For Kubernetes/OpenShift agent deployments, the installation script does not contain the agent software - Docker images containing the agent software will be pulled from the Secure Workload cluster by every Kubernetes/OpenShift node. These connections will be established by the container run time image fetch component and directed at CFG-SERVER-IP:443.

Navigate to **Platform > Cluster Configuration** to know the config server IP and collector IP.

- **Sensor VIP** is for the config server IP: The IP address that has been set up for the config server in this cluster.
- **External IPs** are for collectors IPs and enforcer: If this is populated, when assigning external cluster IP addresses, the selection process is restricted to only IP addresses defined in this list, that are part of the external network.



**Note**

- Secure Workload agent always acts as a client to initiate the connections to the services hosted within the cluster, it will never open a connection as a server.
- In addition to the above permanent connections, for the given agent type that upgrade is supported, agent will periodically perform https requests (port 443) to the cluster sensor VIP to query the available packages.
- Agent is allowed to be located behind a NAT server.

It is important to note that if the workload is behind a firewall or the host firewall service is enabled, then the connections to the cluster might be denied. It is necessary for the administrators to allow such connections by creating appropriate firewall policies.

## Security Exclusions

Software agents continuously interact with the host's operating system during their normal operations. This may cause other security applications installed on the host, such as antivirus, security agents, and others, to raise alarms or block the actions of Secure Workload agents. Therefore, to ensure that agents are installed successfully and are functioning, you must configure the necessary security exclusions on the security applications that are monitoring the host.

**Table 3: Security exclusions for agent directories**

Host OS	Directories
AIX	/opt/cisco/tetration

Host OS	Directories
Linux	/usr/local/tet or /opt/cisco/tetration or <user chosen inst dir>
Windows	C:\Program Files\Cisco Tetration

Table 4: Security exclusions for agent processes

Host OS	Processes
AIX	tet-engine, tet-sensor, tet-enforcer
Linux	tet-engine, tet-sensor, tet-enforcer, tet-main, enforcer
Windows	TetSenEngine.exe, TetSen.exe, TetEnfEngine.exe, TetEnfC.exe, TetEnf.exe, TetUpdate.exe, tet-main.exe

Table 5: Security exclusions for agent actions

Host OS	Actions
AIX	Access /dev/bpf*, /dev/ipl, /dev/kmem, invokes: curl
Linux	Scan /proc, open netlink sockets, invokes: curl, rpm/dpkg, ip[6]tables-save, ip[6]tables-restore, ipset-restore
Windows	Access Registry, register to Firewall Events

Table 6: Security exclusions for agents scripts or binaries executions

Host OS	Invoked scripts/binaries
AIX	ksh: fetch_sensor_id.sh, check_conf_update.sh
Linux	bash: fetch_sensor_id.sh, check_conf_update.sh
Windows	cmd: fetch_sensor_id.cmd, check_conf_update.cmd, dmidecode.exe, npcap-installer.exe, sensortools.exe, signtool.exe

## Service Management of Agents

Software agents are deployed as a service in all supported platforms. This section describes methods to manage the services for various functions and platforms.



**Note** Unless specified otherwise, all the commands in this section require root privileges on Linux or Unix, or administrative privileges on Windows to run.

## Service Management for RHEL, CentOS, OracleLinux-6.x, and Ubuntu-14

Run the following commands for:

- **Starting a service:** `start csw-agent`
- **Stopping a service:** `stop csw-agent`
- **Restarting a service:** `restart csw-agent`
- **Checking service status:** `status csw-agent`

## Service Management for RHEL, CentOS, OracleLinux-7.x and Later

The commands are also applicable to:

- AlmaLinux, Rocky Linux- 8.x and later
- Amazon Linux 2 and later
- Debian 8 and later
- SLES-12SPx and later
- Ubuntu-16.04 and later

Run the following commands for:

- **Starting a service:** `systemctl start csw-agent`
- **Stopping a service:** `systemctl stop csw-agent`
- **Restarting a service:** `systemctl restart csw-agent`
- **Checking service status:** `systemctl status csw-agent`

## Service Management for Windows Server or Windows VDI

Run the following commands for:

- **Starting a service:** `net start <service-name>`  
Example: **net start tetsensor** for deep visibility service - **net start tetenforcer** for enforcement service
- **Stopping a service:** `net stop <service-name>`  
Example: **net stop tetsensor** for deep visibility service - **net stop tetenforcer** for enforcement service
- **Restarting a service:**
  1. `net stop <service-name>`
  2. `net start <service-name>`
- **Checking service status:** `sc query <service-name>`  
Example: **sc query tetsensor** for deep visibility service - **sc query tetenforcer** for enforcement service

## Service Management for AIX

Run the following commands for:

- **Starting a service:** `startsrc -s csw-agent`
- **Stopping a service:** `stopsrc -s csw-agent`
- **Restarting a service:**
  1. `stopsrc -s csw-agent`
  2. `startsrc -s csw-agent`
- **Checking service status:** `lssrc -s csw-agent`

## Service Management for Kubernetes Agent Installations

- **Starting or stopping a service:** It is not possible to start or stop the agents on a specific node because they are not installed as individual services but rather as a clusterwide daemon set.
- **Restarting an agent on a node:** Locate the Secure Workload agent pod on the node and run the appropriate Kubernetes command to kill it. The pod is automatically restarted.
- **Checking the status of pods:** `kubectl get pod -n tetration` OR `oc get pod -n tetration` (for OpenShift) lists the status of all Secure Workload agent pods in the Kubernetes cluster.

## Policy Enforcement with Agents

By default, agents that are installed on your workloads have the capability to enforce policy, but enforcement is disabled. When you are ready, you can enable these agents to enforce policy on selected hosts that are based on the configured intent.

When an agent enforces a policy, it applies an ordered set of rules that specify whether the firewall should ALLOW or DROP specific network traffic that is based on parameters such as the source, destination, port, protocol, and direction. For more information on policies, see [Policies](#).

### Enforcement using agents

- Agents receive policies over a secured TCP or SSL channel.
- Agents run in a privileged domain. On Linux machines, the agent runs as root; on Windows machines, the agent runs as SYSTEM.
- Depending on the platform, when policy enforcement is enabled, agents can completely control the firewall or work with existing configured rules.
- For details about enforcement options and to enable and configure agents to enforce policies, see [Create an Agent Configuration Profile, on page 54](#).

### Advanced details

When you enable enforcement, golden rules are formulated to allow the agent to connect to the controller. Agents communicate with the Enforcement Front End (EFE) of the controller through a bidirectional and

secure channel using the TLS or SSL protocol. Messages from the controller are signed by the policy generator and verified by the agent.

The agent receives policies in a platform-independent schema from the controller. The agent converts these platform-independent policies into platform-specific policies and programs the firewall on the endpoint.

The agent actively monitors the firewall state. If the agent detects any deviation in the enforced policies, it enforces the cached policies into the firewall again. The agent also monitors its own consumption of system resources such as CPU and memory.

The agent periodically sends a status and stats report to the controller using EFE. The status report includes the status of the latest programmed policies such as success, failure, or error, if any. The stats report includes the policy stats such as allowed and dropped packets, and byte count depending on the platform.

## Agent Enforcement on the Linux Platform

On the Linux platform, the agent uses the iptables, ip6tables, or ipset to enforce network policies. After the agent is enabled on the host, by default, it controls, and programs iptables. If the IPv6 network stack is enabled, then the agent controls the IPv6 firewall using ip6tables.

### Linux iptables or ip6tables

The Linux kernel has iptables and ip6tables that are used to set up, maintain, and inspect the tables of IPv4 and IPv6 packet filter rules. The iptables and ip6tables consist of many predefined tables. Each table contains predefined chains and can also contain user-defined chains. These chains contain sets of rules and each of these rules specifies the match criteria for a packet. Predefined tables include raw, mangle, filter, and NAT. Predefined chains include INPUT, OUTPUT, FORWARD, PREROUTING, and POSTROUTING.

The Secure Workload agent programs a filter table that contains rules to allow or drop packets. The filter table consists of the predefined chains INPUT, OUTPUT, and FORWARD. Along with these, the agent adds custom TA chains to categorize and manage the policies from the controller. These TA chains contain Secure Workload rules that are derived from the policies along with rules that are generated by the agent. When the agent receives platform-independent rules, it parses and converts them into iptable, ip6table, or ipset rules and inserts these rules into TA defined chains in the filter table. After programming the firewall, the agent monitors the firewall for any rule or policy deviation and if so, reprograms the firewall. It keeps track of the policies that are programmed in the firewall and reports their stats periodically to the controller.

Here is an example to depict this behavior:

A typical policy in a platform-independent network policy message consists of:

```
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
```

Along with other information, the agent processes this policy and converts it into platform-specific ipset and iptables rule:

```
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
```

## Caveats

### ipset Kernel Module

When enforcement is enabled and preserve rules is disabled in the Agent Config profile, the agents running on Linux hosts ensures that the ipset kernel module has a sufficiently large *max\_sets* configuration. In case a change is needed, the agent reloads the ipset kernel module with a new *max\_sets* value. If Preserve Rules is enabled, the agents check the current ipset module *max\_sets* value, but does not make any change. The current configured *max\_sets* value can be found in `cat /sys/module/ip_set/parameters/max_sets`.

### Host Firewall Backup

The first time that enforcement is enabled in the Agent Config profile, the agents running on Linux hosts, store the current content of ipset and ip[6]tables in `/opt/cisco/tetration/backup` before taking control of the host firewall.

Successive disable or enable transitions of enforcement configuration do not generate backups. The directory is not removed after agent uninstallation.

## Agent Enforcement on the Windows Platform in WAF mode

On the Windows platform, the Secure Workload agent uses the Windows Firewall to enforce network policies.

### Windows Firewall with Advanced Security

A native component on Windows, the Windows Firewall with Advanced Security, regulates network traffic that is based on the following types of settings:

- Rules that regulate inbound network traffic.
- Rules that regulate outbound network traffic.
- Override rules that is based on the authentication status of the source and destination of the network traffic.

- Rules that apply to IPsec traffic and to Windows services.

The Secure Workload Network Policy is programmed using inbound and outbound firewall rules.

## Secure Workload Rules and the Windows Firewall

On the Windows platform, the Secure Workload Network Policy is enforced as follows:

1. The platform-independent firewall rules from the Secure Workload Network policy are translated into Windows Firewall rules.
2. The rules are programmed in Windows Firewall.
3. The Windows Firewall enforces the rules.
4. The Windows Firewall and its ruleset are monitored. If a change is detected, the deviation is reported and the Secure Workload Network policy is reset in the Windows Firewall.

## Security Profiles

Windows Firewall groups the rules based on the network that the host is connected to. These rule groups are called Profiles and there are three such profiles:

- Domain Profile
- Private Profile
- Public Profile

The Secure Workload rules are programmed into all the profiles, but only rules in active profiles are continuously monitored.

## Effective Setting and Mixed-List Policies

The set of rules in the Windows Firewall is not ordered based on the precedence. When multiple rules match a packet, the most restrictive of those rules take effect meaning that DENY rules take precedence over ALLOW rules. For more information, see the article on [Microsoft TechNet](#).

Consider the mixed-list, both allow and deny, policy example from the Enforcement Agent section:

```
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
```

When a packet headed for the host 1.2.3.30 TCP port 80 reaches the firewall, it matches all the rules above, but the most restrictive of them all, Rule number 3, is the one that will be enforced and the packet will be dropped. This behavior is contrary to the expectation that the rules will be evaluated in order, Rule 1 will be the rule that is enforced, and that the packet will be allowed.

This difference in behavior is expected in the Windows platform owing to the design of the Windows Firewall described above. This behavior can be observed in mixed-list policies with overlapping rules that have different rule actions.

For example,

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

### Interference from Other Firewalls or Policies

We recommend that you grant the agent full and exclusive control of the Windows Firewall to enforce the Secure Workload Network Policy as intended. Agents cannot reliably enforce the policy if:

- A third-party firewall is present. (The Windows Firewall is required to be the active firewall product on the host.)
- The Firewall is disabled for the current profiles.
- Conflicting firewall settings are deployed using Group Policy. Some of the conflicting settings are:
  - Firewall rules.
  - Default inbound or outbound actions in the current profiles that differ from the catch-all rules of the policy.
  - Firewall disabled for the current profiles.

## Stateful Enforcement

Windows Advanced Firewall is considered as a **stateful** firewall, that is for certain protocols such as TCP, the firewall maintains internal state tracking to detect if a new packet hitting the firewall belongs to a known connection. Packets belonging to a known connection are allowed without the firewall rules having to be examined. A stateful firewall enables bidirectional communication without rules having to be established in the INBOUND and OUTBOUND tables.

For example, consider the following rule for a web server: **Accept all TCP connections to port 443**

The intention is to accept all TCP connections on port 443 to the server, and allow the server to communicate back to the clients. In this case, only one rule is inserted in the INBOUND table, allowing TCP connections on port 443. No rule is required to be inserted in the OUTBOUND table. Inserting a rule in the OUTBOUND table is implicitly done by the Windows Advanced Firewall.




---

**Note** Stateful tracking applies only to protocols that establish and maintain explicit connections. For other protocols, both INBOUND and OUTBOUND rules must be programmed to enable bidirectional communication.

---

When enforcement is enabled, a given concrete rule is programmed as **stateful** when the protocol is TCP (the agent decides, based on the context, whether the rule is to be inserted in the INBOUND table or the OUTBOUND table). For other protocols (including **ANY**), both INBOUND and OUTBOUND rules are programmed.

## Caveats

### Host firewall backup

When enforcement is enabled for the first time in the Agent Config profile, the agents running on Windows hosts, before taking control of the host firewall, export the current Windows Advanced Firewall content to `ProgramData\Cisco\Tetration\backup`. Successive disable or enable transitions of Enforcement configuration do not generate backups. The directory is not removed upon agent uninstallation.



## Agent Enforcement on the Windows Platform in WFP Mode

On the Windows platform, the agent enforces the network policies by programming Windows Filtering Platform (WFP) filters. Windows Advanced Firewall is not used to configure the network policy.

### Windows Filtering Platform

Windows Filtering Platform (WFP) is a set of APIs provided by Microsoft to configure filters for processing network traffic. Network traffic processing filters are configured using kernel-level APIs and user level APIs. WFP filters can be configured at various layers, Network Layer, Transport Layer, Application Layer Enforcement (ALE). Secure Workload WFP filters are configured at the ALE layer, similar to Windows firewall rules. Each layer has several sublayers, ordered by weight, from highest to lowest. Within each sublayer, filters are ordered by weight, from highest to lowest. A network packet traverses through all the sublayers. At each sublayer, the network packet traverses through the matching filters that are based on weight, from highest to lowest and returns the action: Permit or Block. After passing through all the sublayers, the packet is processed based on the rule that Block action overrides Permit.

### Advantages of WFP over WAF

- Avoids Windows Firewall configuration dependencies.
- Overcomes GPO restrictions.
- Ensures ease of migration and policy reversion.
- Allows you to control policy ordering.
- Avoids strict block-first policy order of Windows Firewall.
- Reduces CPU overhead on policy update.
- Creates an efficient 1:1 policy rule filter.
- Ensures a faster single-step update.

### Agent Support for WFP

When enforcement is configured to use WFP, Secure Workload filters override Windows Firewall rules.

In WFP mode, the agent configures the following WFP objects:

- Provider has a GUID and name, is used for filter management, and does not affect packet filtering
- Sublayer has a GUID, name, and weight. The Secure Workload sublayer is configured with higher weight than the Windows Advanced Firewall sublayer.
- Filter has name, GUID, ID, weight, layer ID, sublayer key, action (PERMIT/ BLOCK), and conditions. WFP filters are configured for Golder rules, Self Rules, and Policy Rules. The agent also configures the port scanning prevention filters. Secure Workload filters are configured with the FWPM\_FILTER\_FLAG\_CLEAR\_ACTION\_RIGHT flag. This flag ensures that Secure Workload filters are not overridden by Microsoft Firewall rules. For each Secure Workload Network policy rule, one or more WFP filters are configured based on the direction (inbound or outbound) and protocol.

For TCP inbound policy,

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

The WFP filters configured are:

```
Filter Name:                Secure Workload Rule 14
-----
EffectiveWeight:           18446744073709551589
LayerKey:                  FWPM_LAYER_ALE_AUTH_LISTEN_V4
Action:                    Permit
Local Port:                3389
Filter Name:                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
```

Secure Workload agent configures **Secure Workload Default Inbound** and **Secure Workload Default Outbound** filters for inbound and outbound CATCH-ALL policy respectively.

## Agent WFP support and Windows Firewall

- The agent **does not monitor** WAF rules or WAF profiles.
- The agent **does not monitor** firewall states.
- The agent **does not require** firewall state to be enabled.
- The agent **does not conflict** with GPO policies.

## Effective Setting and Mixed-List Policies

Agent enforcement in WFP mode supports mixed-list or grey list policies.

Consider the mixed-list (both allow and deny) policy example from the Enforcement Agent section:

```
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, ALLOWegress - wt996
```

When a packet headed for the host 1.2.3.30 tcp port 80 reaches the firewall, it matches filter 1 and is allowed. However, a packet that is headed for the host 1.2.3.10 is blocked because of filter 2. A packet that is headed for host 1.2.2.10 is allowed by filter 3.

## Stateful Enforcement

Secure Workload WFP filters are configured at the ALE layer. Network traffic is filtered for socket connect(), listen(), and accept() operations. Network packets related to a L4 connection are not filtered after the connection is established.

## Visibility of Configured WFP Filters

You can view the configured Secure Workload WFP filters using `c:\program files\tetration\tetenf.exe`. Supported options:

- With administrative privileges, run `cmd.exe`.
- Run `c:\program files\tetration\tetenf.exe -l -f <-verbose> <-output=outfile.txt>`.

OR

- With administrative privileges, run `cmd.exe`.
- Run `netsh wfp show filters`.
- View the `filters.xml` file for configured Secure Workload filters.

## Disable Stealth Mode Filters in WFP Mode

To disable stealth mode filters (port scanning filters):

- 
- Step 1** Edit `\conf\enforcer.cfg`.
- Step 2** Add `disable_wfp_stealth_mode: 1`
- Step 3** Save the file.
- Step 4** With administrative privileges, restart the `tetenforcer` service by:
- Run the command: `sc stop tetenforcer` to stop TetEnforcer Service.
  - Run the command: `sc start tetenforcer` to start TetEnforcer Service.
- Step 5** To verify:
- With administrative privileges, run `cmd.exe`.
  - Run the command: `c:\program files\tetration\tetenf.exe -l -f <-verbose> <-output=outfile.txt>`.
- 

`"Tetration Internal Rule block portscan" filters are not configured.`

## Delete Configured WFP Filters

You can delete the configured Secure Workload WFP filters using `c:\program files\tetration\tetenf.exe`. To avoid accidental deletions of filters, when you run the delete command, specify the token in `<yyyymm>` format, where `yyyy` is the current year and `mm` is the current month in the numerical form. For example, if today's date is 01/21/2021, then the token is **-token=202101**

Supported options are:

- With administrative privileges, run `cmd.exe`.
- To delete all configured Secure Workload filters, run `c:\program files\tetration\tetenf.exe -d -f -all -token=<yyyymm>`
- To delete all configured Secure Workload WFP objects, run `c:\program files\tetration\tetenf.exe -d -all -token=<yyyymm>`
- To delete a Secure Workload WFP filter by name, run `c:\program files\tetration\tetenf.exe -d -name=<WFP filter name> -token=<yyyymm>`

## Known Limitations in WFP Mode

- The **Preserve Rules** setting in Agent Config Profile has no effect when you set Enforcement Mode to WFP.

## Configure Policies for Windows Attributes

For more granularity when enforcing a policy on Windows-based workloads, you can filter network traffic by:

- Application Name
- Service Name
- User Names with or without User Groups

This option is supported in both WAF and WFP modes. Windows OS-based filters are categorized as *consumer filters* and *provider filters* in the generated network policy. The Consumer filters filter the network traffic that is initiated on the consumer workload and Provider filters filter the network traffic that is destined for the provider workload.

### Before you begin

This procedure assumes you are modifying an existing policy. If you have not yet created the policy to which you want to add a Windows OS-based filter, create that policy first.

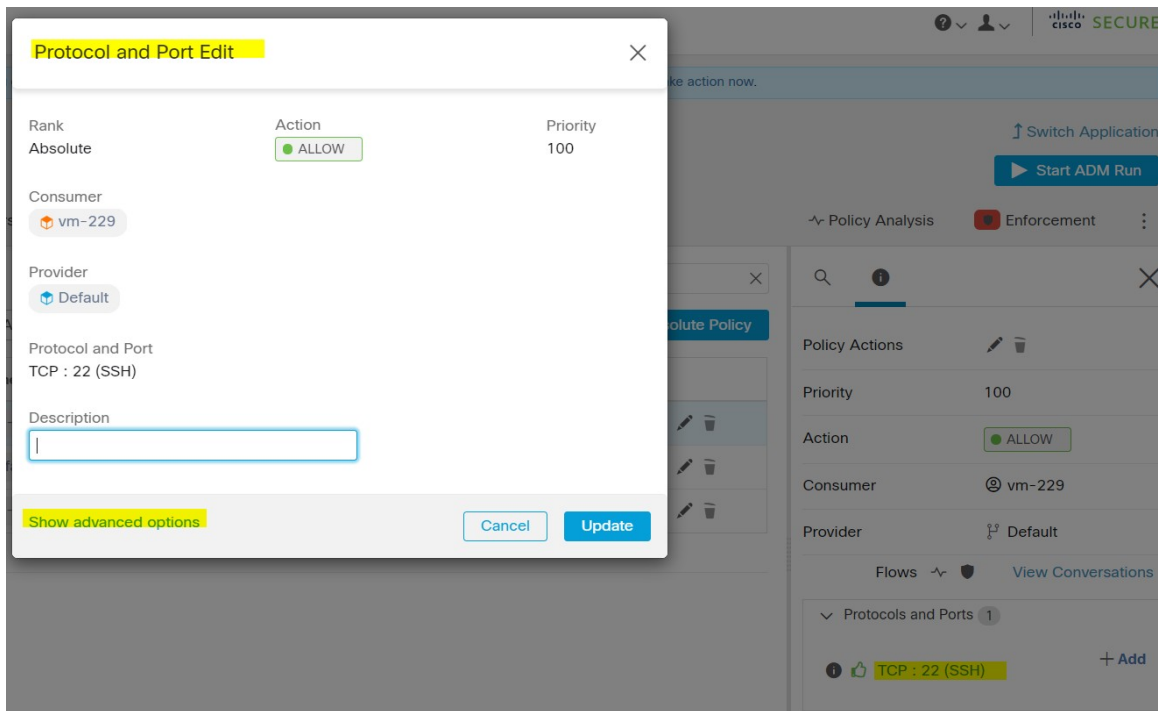



---

**Important** See [Caveats, on page 39](#) and [Known limitations, on page 39](#) for policies involving Windows attributes.

---

- 
- Step 1** In the navigation pane, click **Defend > Segmentation**.
- Step 2** Click the scope that contains the policy for which you want to configure Windows OS-based filters.
- Step 3** Click the workspace in which you want to edit the policy.
- Step 4** Click **Manage Policies**.
- Step 5** Choose the policy to edit.
- Important** Consumer and Provider must include only Windows workloads.
- Step 6** In the table row for the policy to edit, click the existing value in the **Protocols and Ports** column.
- Step 7** In the pane on the right, click the existing value under **Protocols and Ports**.  
In the example, click **TCP : 22 (SSH)**.



**Step 8** Click **Show advanced options**.

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 ⓘ

Provider Service

Provider Binary Path

Provider Users or User Groups ⓘ

[Hide advanced options](#) Cancel

**Step 9** Configure consumer filters based on Application name, Service name, or User name.

- The application name must be a full pathname.
- Service name must be a short service name.

- User name can be a local user name (For example, `tetter`) or domain user name (For example, `sensor-dev@sensor-dev.com` or `sensor-dev\sensor-dev`)
- User group can be local user group (For example, `Administrators`) or domain user group (For example, `domain users\sensor-dev`)
- Multiple user names and/ or user group names can be specified, separated by `,"`. (For example, `sensor-dev@sensor-dev.com,domain users\sensor-dev`)
- Service name and User name cannot be configured together.

**Step 10** Configure provider filters based on Application name, Service name, or User name.

Follow the same guidelines as given for consumer filters in the previous step.

**Step 11** Enter the paths to the binary, as applicable.

For example, enter `c:\test\putty.exe`

**Step 12** Click **Update**.

## Recommended Windows OS-Based Policy Configuration

Always specify ports and protocols in policies when possible; we recommend not to allow ANY port, ANY protocol.

For example, a generated policy with port and protocol restrictions might look like this:

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

In contrast, if you allow network connections that are initiated by `iperf.exe` with ANY protocol and ANY port, the generated policy looks like this:

```
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:"
}
```

For the above filter, Secure Workload creates a policy rule to allow the network traffic on the provider as follows:

```

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

```

This network rule opens all the ports on the Provider. We strongly recommend not to create OS-based filters with *Any* protocol.

## Known limitations

- Windows 2008 R2 does not support Windows OS based filtering policies.
- Network policy can be configured with a single user name whereas MS Firewall UI supports multiple users.

## Caveats

- While using the Windows OS-based policies, a consumer/ provider scope or filter should only contain Windows agents. Otherwise, non-Windows OSs (Linux, AIX) skip the policy and report a sync error in Enforcement Status.
- Avoid creating Windows OS filters with *loose* filtering criteria. Such criteria may open unwanted network ports.
- Due to limited or no knowledge of the process context, user context or service context of the network flows, there will be discrepancy in the policy analysis if the policies have Windows OS-based filters.

## Verify and Troubleshoot Policies with Windows OS-Based Filtering Attributes

If you use Windows OS-based filtering attributes, the following topics provide you with verification and troubleshooting information.

Cisco TAC can use this information as needed to troubleshoot such policies.

### Policies Based on Application Name

Use the following information to verify and troubleshoot policies based on application name on Windows OS workloads.

The following sections describe the way policies should appear on the workload for an application binary entered as `c:\test\putty.exe`.

#### Sample Policy Based on Application Name

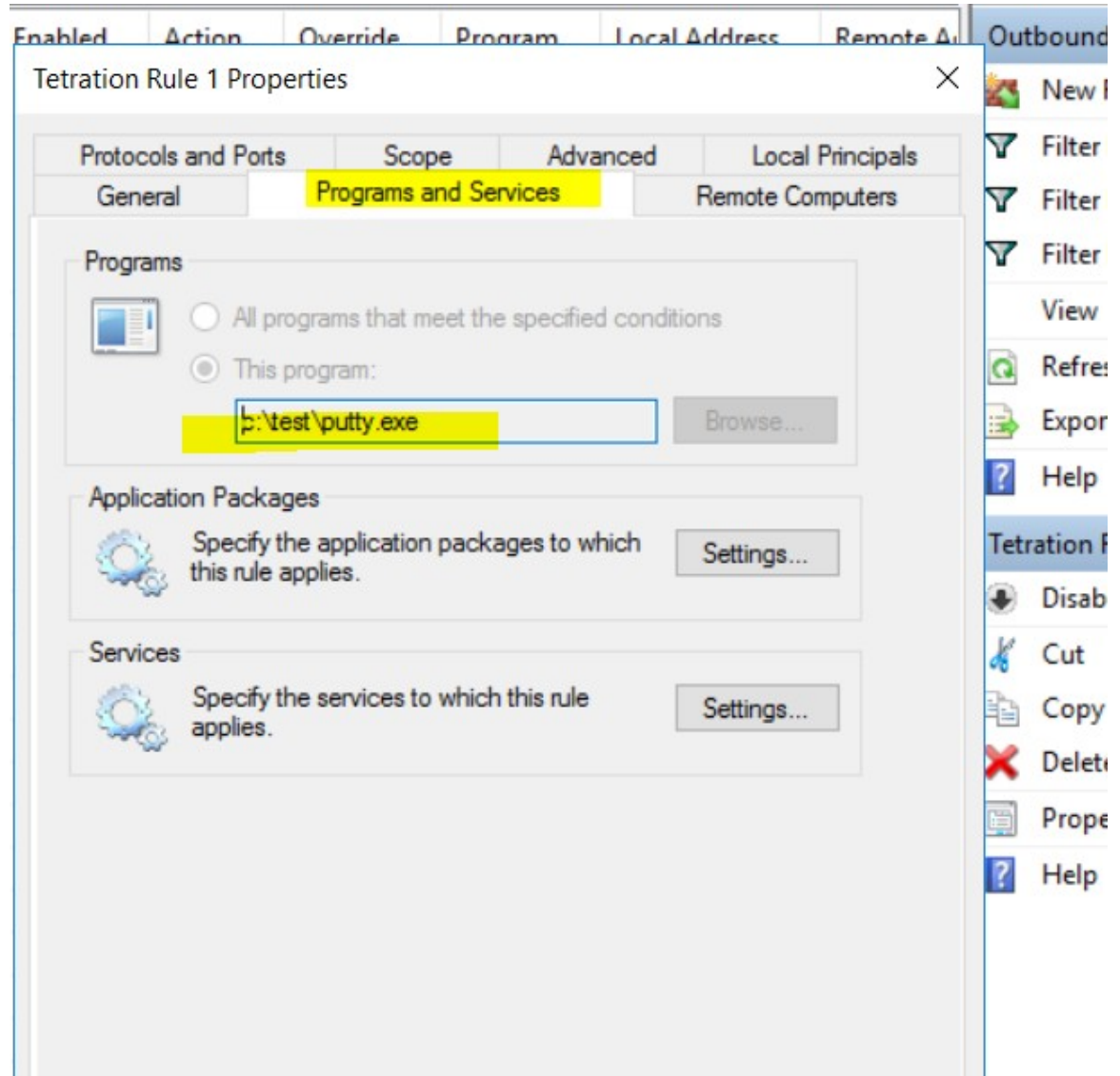
```

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
```

### Generated Firewall Rule



### Generated Filter Using netsh

To verify, using native Windows tools, that a filter has been added to an advanced policy:

- With administrative privileges, run `cmd.exe`.
- Run `netsh wfp show filters`.
- The output file, **filters.xml**, is generated in the current directory.
- Check `FWPM_CONDITION_ALE_APP_ID` for the application name in the output file: `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>

```

### Generated WFP Filter Using `tetenf.exe -l -f`

```

Filter Name:          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

```

### Invalid Application Name

- In WAF mode, Firewall rule is created for an invalid application name.
- In WFP mode, the WFP filter is not created for an invalid application name but the NPC is not rejected. The agent logs a warning message and configures the rest of the policy rules.

## Policies Based on Service Name

Use the following information to verify and troubleshoot policies based on Service name on Windows OS workloads.

The following sections describe the way that the policies should appear on the workload.

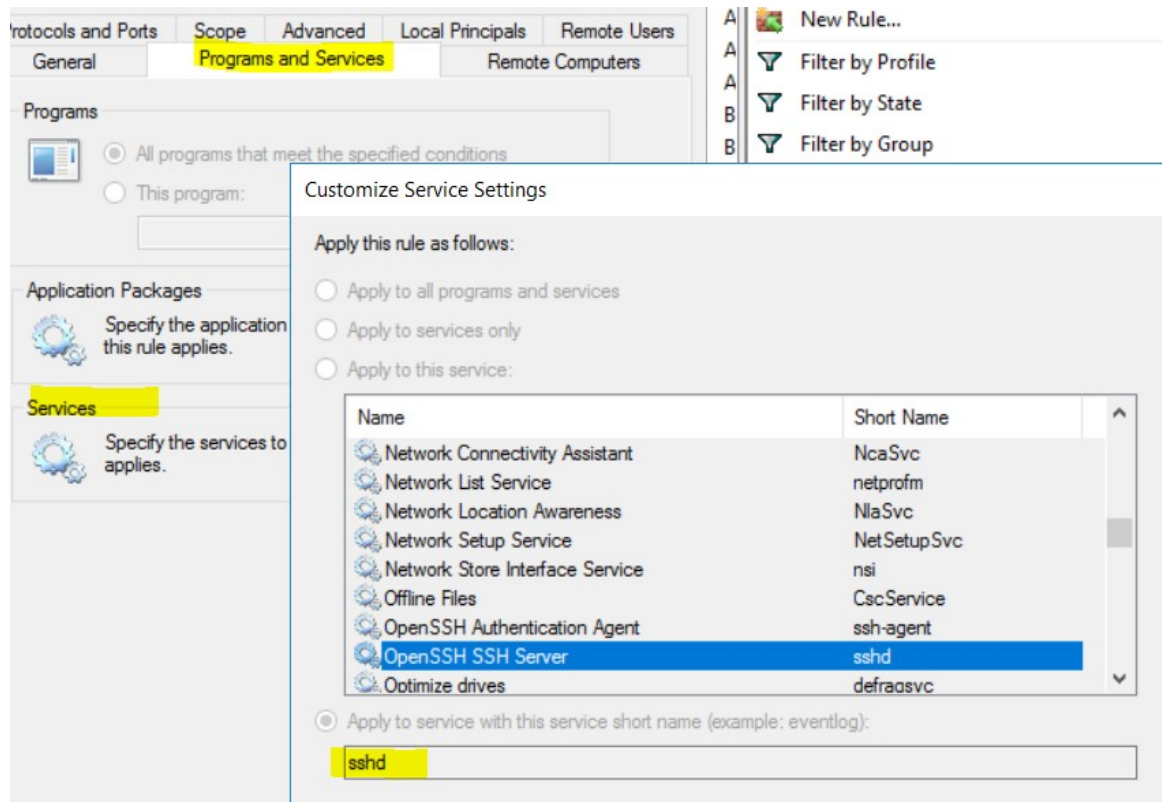
### Sample Policy Based on Service Name

```

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

```

## Generated Firewall Rule



## Generated Filter Using netsh

To verify using native Windows tools, that a filter has been added for an advanced policy:

- With administrative privileges, run `cmd.exe`.
- Run `netsh wfp show filters`.
- The output file, **filters.xml**, is generated in the current directory.
- Check `FWPM_CONDITION_ALE_USER_ID` for user name in the output file: `filters.xml`.

```
<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>
```

## Generated WFP Filter Using `tetentf.exe -l -f`

```
Filter Name:          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
```

### Invalid Service Name

- In WAF mode, the Firewall rule is created for a nonexistent service name.
- In WFP mode, the WFP filter is not created for a nonexistent service name.
- Service SID type must be *Unrestricted* or *Restricted*. If the service type is *None*, the Firewall Rule and WFP filter can be added but they have no effect.

To verify the SID type, run the following command:

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

### Policies Based on User Group or User Name

Use the following information to verify and troubleshoot policies based on user name (with and without user group name) on Windows OS workloads.

Sections in this topic describe the way that the policies should appear on the workload.

Examples in this topic are based on policies that are configured with the following information:

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 ⓘ

Provider Service

Provider Binary Path

Provider Users or User Groups ⓘ

### Sample Policy Based on User Name

```
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
```

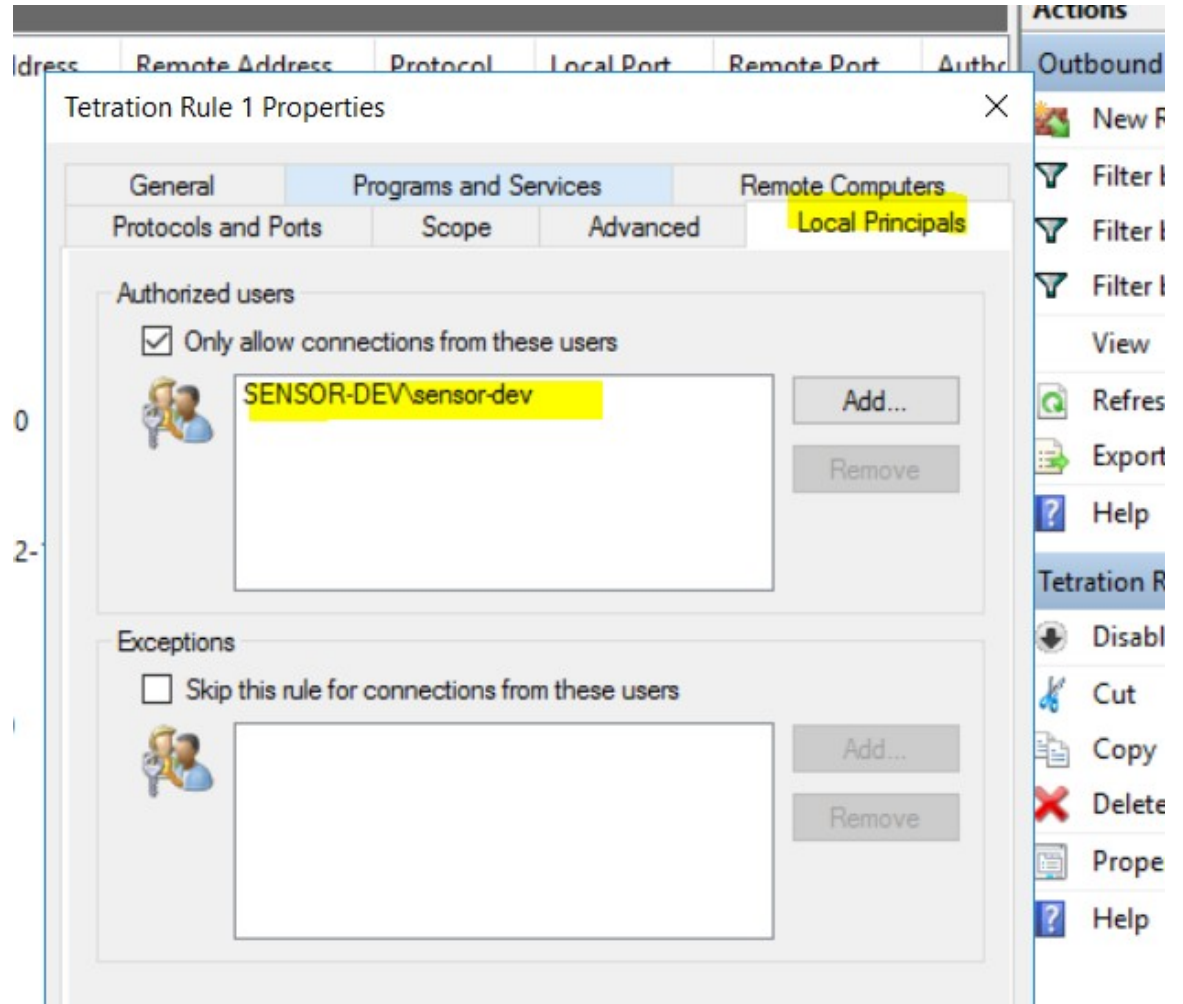
### Sample Policy Based on User Group and User Name

```
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
```

## Generated Firewall Rule

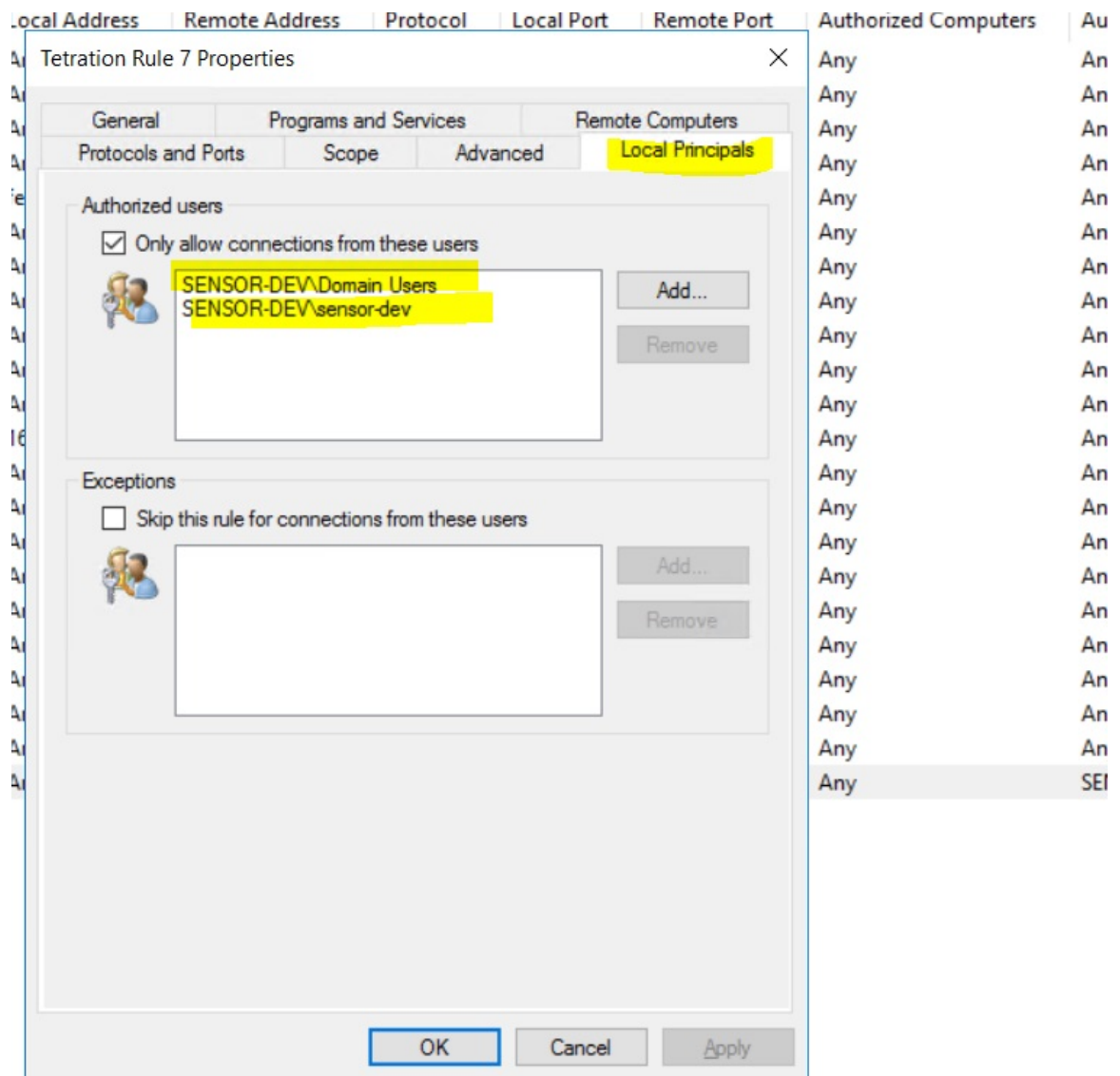
### Firewall Rule Based on User Name

Example: Firewall rule based on User Name, sensor-dev\\sensor-dev



### Firewall Rule Based on User Group and User Name

Example: Firewall rule based on User Name, sensor-dev\\sensor-dev and user group, domain users\\sensor-dev



### Generated Filter Using netsh

To verify using native Windows tools that a filter has been added for an advanced policy:

- With administrative privileges, run `cmd.exe`.
- Run `netsh wfp show filters`.
- The output file, **filters.xml**, is generated in the current directory.
- Check `FWPM_CONDITION_ALE_USER_ID` for user name in the output file: `filters.xml`.

```
<item>
  <fieldKey>FWPM_CONDITION_ALE_USER_ID</fieldKey>
  <matchType>FWP_MATCH_EQUAL</matchType>
  <conditionValue>
    <type>FWP_SECURITY_DESCRIPTOR_TYPE</type>
```

```

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

```

### Generated WFP Filters Using tetenf.exe -l -f

#### Filter based on User Name

Example: WFP Rule based on User Name, SENSOR-DEV\sensor-dev

```

Filter Name:                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

```

#### Filter based on User Group and User Name

Example: WFP Rule based on User Name, SENSOR-DEV\sensor-dev and User Group name, SENSOR-DEV\Domain Users

```

Filter Name:                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

```

*Service name and user name cannot be configured for a Network policy rule.*



**Note** The network policy is rejected by the Windows agent if the user name or the user group is invalid.

## Enforcement of Kubernetes Pods on Windows Nodes

After you install the Kubernetes DaemonSet agent on the Windows worker nodes, it captures the network flows from the Windows worker nodes and the Kubernetes pods in an AKS environment.

### Requirements

- Enforcement of Kubernetes pods is supported in an AKS environment with Windows nodes.
- Enforcement mode MUST be WFP with **Preserve Rules** set to Off.
- Supported on Microsoft Windows Server 2019 and Windows Server 2022.

The policies are enforced on vSwitch for ports that are connected to pods using VFP. The Virtual Filtering Platform (VFP) is a component of vSwitch used to configure filters for processing network traffic. While enforcing the policies, the Preserve Mode is Off.

Each filter has the following attributes:

- Id: Filter Name
- Direction: In or Out
- RuleType: Switch or Host.
  - Configure the filter on vSwitch when the type is Switch.
  - Create a WFP filter when the type is Host.
- Action: Allow or Block
- LocalPorts: This can be a port or range. For example, 80 or 100-200.
- RemotePorts: Same as LocalPorts.
- LocalAddresses: It is an address or range. For example, 10.224.0.5, 10.224.1.0/24 (10.224.1.1-10.224.1.10 is not allowed).
- RemoteAddress: Same as LocalAddresses
- Protocol: ICMP/TCP/UDP/IGMP Protocol 255 is IPPROTO\_RAW and 256 – PROTO\_MAX

The ports can only be specified for UDP and TCP, and ports are not allowed in the policy unless a protocol is specified.

Configuring a policy on a virtual port is a transaction-based operation. If one of the filters is invalid, enforcing the entire policy is rendered unsuccessful.

This is the stateful enforcement. Application, user, or service-based policies are currently not supported.

### Compatibility with Calico

Pods enforcement works in "preserve rules" off mode. When the Windows agent enforces the rules on pods, it deletes the already configured policies. If the Calico plug-in enforces the network policies after the agent, the agent identifies it as **deviation** and network policies that are configured by Calico are deleted and agent policies are re-enforced.




---

**Note** The enforced policies are deleted when the Windows agent is uninstalled on the Windows nodes.

---

### Visibility of Configured VFP Filters

An option to list the pod filters using Secure Workload is not available. In an AKS environment, you can use the built-in PowerShell script. Run the following PowerShell script: `c:\k\debug\collectlogs.ps1`. View the output files **vfoutput.txt** and **hnsdiag.txt** for the configured filters.

### Delete VFP Filters Configured by Windows Agent

1. Run **cmd.exe** with administrative privileges.
2. Run the command: `<installation folder>\tetenf.exe -d -f -pods -token=<yyyyymm>`.





**Note** The command deletes VFP filters for all the pods.

### Troubleshoot Enforced Policies and Network Flows

1. Run the command: `netsh wfp start capture keywords=19`.
2. Run network traffic.
3. Stop capturing the flows: `netsh wfp stop capture`.
4. Extract **wfpdiag.xml** from the **wfpdiag.cab** file. View the dropped flows.

To map the allowed or dropped network flows to Pod policies:

1. Start ETW session: `logman start <session name> -p Microsoft-Windows-Hyper-V-VfpExt -o <output file.etl> -ets`
2. Run network traffic.
3. Stop capturing flows: `logman stop <session name>`.
4. In the command prompt, run: `tracert <output file.etl>`. The command creates the **dumpfile.xml** file. View the network flows.

## Agent Enforcement on AIX Platform

On the AIX platform, the Secure Workload agent uses IPFilter utilities to enforce network policies. By default, after the agent is enabled on the host, the agent controls and programs the IPv4 filter table. IPv6 enforcement is not supported.

### IPFilter

The IPFilter package on AIX is used to provide firewall services and is available on AIX as a kernel expansion pack. It loads as a kernel extension module, `/usr/lib/drivers/ipf`. It includes `ipf`, `ippool`, `ipfstat`, `ipmon`, `ipfs`, and `ipnat` utilities that are used to program ipfilter rules and each of these rules specifies the match criteria for a packet. For more information, see the IPFilter pages in the AIX manual.

When enforcement is enabled, the agent uses IPFilter to program the IPv4 filter table that contains rules for allowing or dropping of IPv4 packets. The agent groups these rules to categorize and manage the policies using the controller. These rules include Secure Workload rules that are derived from the policies and rules that are generated by the agent.

When an agent receives platform-independent rules, it parses and converts them into ipfilter or ippool rules and inserts these rules into the filter table. After programming the firewall, the enforcement agent monitors the firewall for any rule or policy deviation and if so, reprograms the firewall. The agent keeps track of the policies that are programmed in the firewall and reports their status periodically to the controller.

A typical policy in a platform-independent network policy message consists of:

```
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

```

Along with other information, the agent processes the policy and converts it into platform-specific ipool and ipfilter rule:

```

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

```

## Caveats

### Host Firewall Backup

When enforcement is enabled for the first time in an Agent Config Profile, the agents running on AIX hosts, before taking control of the host firewall, store the current content of ipool and ipfilter into */opt/cisco/tetration/backup*. Successive disable or enable transitions of enforcement configuration do not generate backups. The directory is not removed upon agent uninstallation.

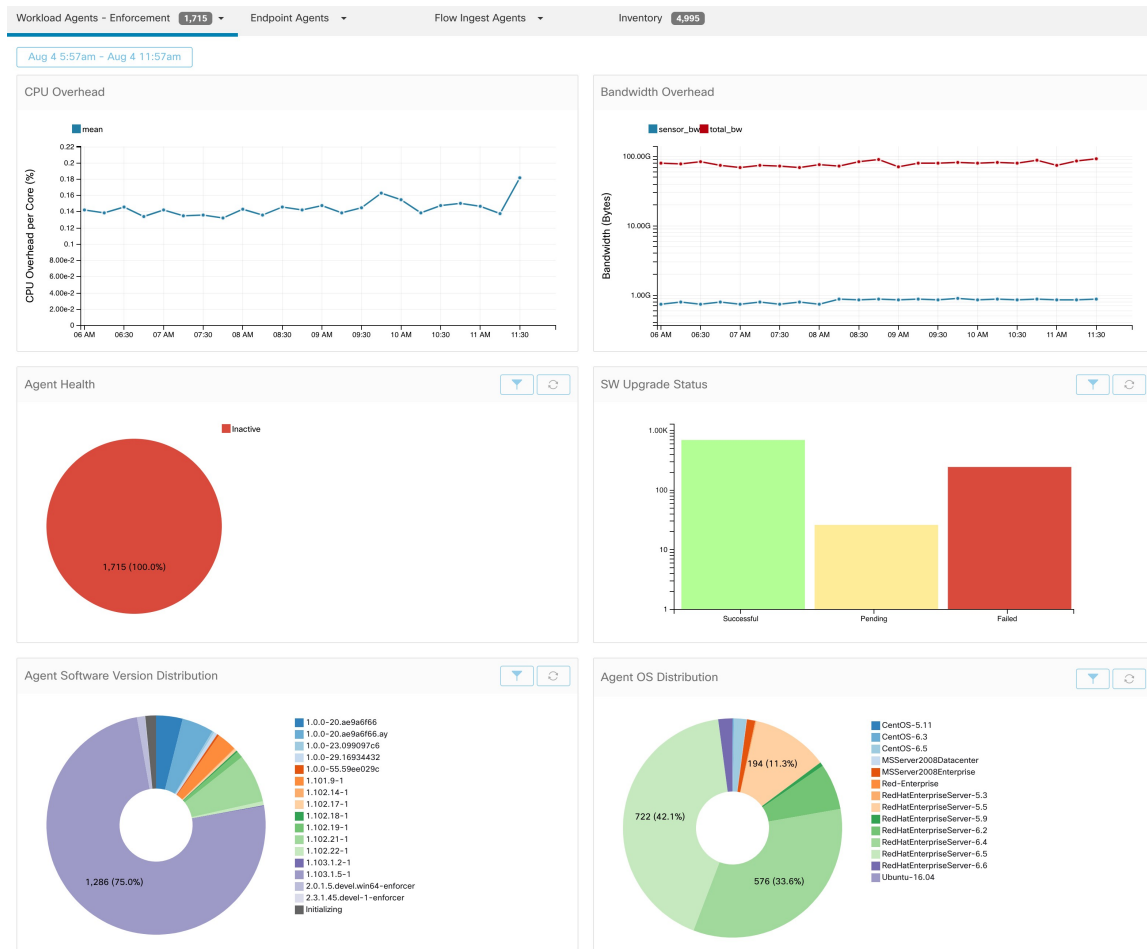
## Known Limitations

IPv6 enforcement is not supported.

## Check Agent Status and Statistics

- 
- Step 1** In the navigation pane, click **Manage > Workloads > Agents**.
  - Step 2** Click the **Distribution** tab.
  - Step 3** Click an agent type from the top of the page.
  - Step 4** On this page, you can check CPU Overhead, Bandwidth Overhead, Agent Health, Software Update Status, Agent Software Version Distribution, and Agent OS Distribution.

Figure 2: Agent Distribution Page



For more information on the enforcement status, see the Enforcement Status section.

## View Agent Details

The following steps provide one of the available options to navigate to the Workload Profile page, which displays details about the workload and its installed agent.

- Step 1** In the navigation pane, click **Organize >> Scopes and Inventory**.
- Step 2** Search for a workload for which you want to view details.
- Step 3** Click the IP address to view the details such as agent health, IP address, Scopes, Inventory Type, Enforcement Groups, Experimental Groups, User Labels, and Traffic Volume (Total Bytes/Total Packets).

For more information, see [Workload Profile](#).

# Software Agent Config

## Requirements and Prerequisites for Configuring Software Agents

- Ensure that you have the required Secure Workloaduser role credentials:
  - Site Administrator
  - Customer Support

For more information, see [User Roles and Access to Agent Configuration, on page 52](#).

- Ensure that you have privileges on the host to run the agent service on each workload. For more information, see [Service Management of Agents](#).
- Verify the supported platforms, requirements, and installation instructions for agents. For more information, see [Deploy Software Agents](#).

## User Roles and Access to Agent Configuration

1. Root scope owners have access only to create a Configuration Profile and Configuration Intent specification.
2. Root scope owners can create configuration profiles that are associated with only owned scopes and impose these configuration profiles on agents that are owned filters or scopes.



**Note** Under the Agent Configuration Profile, you can now see the number of intents using the configuration profile before you edit the profile.

**Figure 3: Software Agent Configuration Tab for Scope Owner Users**

The screenshot displays the 'Configure' tab of the Software Agent Configuration interface. It features a navigation bar at the top with tabs for 'Installer', 'Upgrade', 'Convert to Enforcement Agent', 'Configure' (active), 'Monitor', 'Distribution', and 'Agent List'. Below the navigation bar, there are three main sections:

- Agent Config Profiles:** A table with columns 'Name', 'Config', and 'Actions'. It lists two profiles: 'Enforcement' and 'Default'. The 'Enforcement' profile includes settings like 'Enforcement', 'Windows Enforcement Mode - WAF', 'Preserve Rules', 'Allow Broadcast', 'Allow Multicast', 'Allow Link Local Addresses', 'CPU Quota Mode - Adjusted (3%)', and 'Memory Quota Limit - 512MB'. The 'Default' profile includes settings like 'Flow Analysis Fidelity - Detailed', 'Data Plane', 'Auto-Upgrade', 'PID Lookup', 'CPU Quota Mode - Adjusted (3%)', 'Memory Quota Limit - 512MB', 'Process Visibility and Forensics', 'Forensics', 'Meltdown Exploit Detection', 'CPU Quota Mode - Adjusted (3%)', and 'Memory Quota Limit - 256MB'. An 'Edit' button is visible next to the 'Default' profile.
- Agent Config Intents:** A section with a filter dropdown set to 'Default' and buttons for 'Create Intent', 'Edit', and 'Delete'. It also includes a 'View Deleted Agent Config Intents' link and a 'No intents found' message.
- Agent Remote VRF Configurations:** A section with a 'Create Config' button and a 'No configs found' message.

3. Site administrators have access to all the components in the Agent Configuration page that include specifying interface configuration intents and remote virtual routing and forwarding configurations.

Figure 4: Software Agent Configuration Tab for Site Administrators

The screenshot displays the 'Configure' tab of the Software Agent Configuration interface. It features a top navigation bar with tabs: Installer, Upgrade, Convert to Enforcement Agent, **Configure**, Monitor, Distribution, and Agent List. The main content area is divided into four sections:

- Agent Config Profiles:** A table with columns for Name, Config, and Actions. It lists two profiles: 'Default' and 'VM'. Each profile has a list of configuration items with status indicators (green for enabled, red for disabled). For example, 'Enforcement' is enabled in both, while 'Windows Enforcement Mode - WAF' is disabled in 'Default' but enabled in 'VM'. The 'Default' profile has an 'Edit' button, while the 'VM' profile has 'Edit' and 'Delete' buttons.
- Agent Config Intents:** A section with a 'Create Intent' button. It shows two intents: 'Apply profile Default to filter Tetration' (with 'Edit' and 'Delete' buttons) and 'Apply profile Default to filter Everything'.
- Interface Config Intents:** A section with a 'Create Intent' button. It shows one intent: 'Apply VRF Default to filter Tetration' (with 'Edit' and 'Delete' buttons).
- Agent Remote VRF Configurations:** A section with a 'Create Config' button. It currently displays 'No configs found'.

## Configure Software Agents

On the Software Agent Configuration page, configure the software agents to create intents that are associated with either an **Inventory Filter** or a **Scope**. For each agent, apply the first matching intent. For more information, see [Inventory](#).



**Note** For any Secure Workload deployment, a default agent configuration is applied to all agents that are not associated with any specific configuration profile.

Figure 5: Software Agent Configuration

The screenshot displays the 'Configure' tab of the Software Agent configuration interface. It is divided into two main sections: 'Agent Config Profiles' and 'Agent Config Intents'.

**Agent Config Profiles:** This section contains a table with columns for 'Name', 'Config', and 'Actions'. There are two profiles listed: 'Default' and 'VM'. Each profile has a list of configuration options with status indicators (green for enabled, red for disabled). The 'Default' profile has 'Enforcement' and 'Flow Visibility' enabled, while 'VM' has 'Enforcement' disabled. Both profiles have 'Data Plane', 'Auto-Upgrade', and 'Memory Quota Limit' enabled. The 'Process Visibility and Forensics' section is disabled in both.

**Agent Config Intents:** This section contains three panels:
 

- Apply profile Default to filter Tetration:** Includes 'Edit' and 'Delete' buttons.
- Apply profile Default to filter Everything:** A simple text-based intent.
- Interface Config Intents:** Includes a 'Create Intent' button and a panel for 'Apply VRF Default to filter Tetration' with 'Edit' and 'Delete' buttons.

At the bottom of the 'Agent Config Profiles' section, there is a 'View Deleted Agent Config Profiles' link.

## Create an Agent Configuration Profile

### Before you begin

See [Requirements and Prerequisites for Configuring Software Agents](#), on page 52.

- Step 1** In the navigation bar, click **Manage > Workloads > Agents**.
- Step 2** Click the **Configure** tab.
- Step 3** Click the **Create Profile** button.
- Step 4** Enter a name for the profile (required) and choose the scopes where the profile is available.
- Step 5** Enter the appropriate values in the fields listed in the table:

**Table 7: Enforce Configuration**

Option	Description
<b>Enforcement</b>	<p><b>Enable</b> - Enable policy enforcement on the agent. After you enable enforcement, the agent enforces the most recently received policy set. <b>Disable</b> (Default) - The agent does not enforce a policy.</p> <p><b>Note</b> If you enable, disable and re-enable policy enforcement on the agent, it clears the firewall state and sets the catch-all default action to ALLOW.</p>

Option	Description
<b>Preserve Rules</b>	<p><b>Enable</b> - Preserves existing firewall rules on the agent.</p> <p><b>Disable (Default)</b> - Clears existing firewall rules before applying enforcement policy rules from Secure Workload.</p> <p>Behaviour of the Preserve Rules attribute is platform-specific. You can view the details of the attributes in the Preserve Rules section in each platform.</p>
<b>Allow Broadcast</b>	<p><b>Enable (Default)</b> - Adds rules to the firewall to allow ingress and egress broadcast traffic on workload.</p> <p><b>Disable</b> - Does not add any rules. The broadcast traffic drops if the default policy on the agent is DENY.</p>
<b>Allow Multicast</b>	<p><b>Enable (Default)</b> - Adds rules to the firewall to allow ingress and egress multicast traffic on workload.</p> <p><b>Disable</b> - Does not add any rules. The Multicast traffic drops if the default policy on the agent is DENY.</p>
<b>Allow Link Local</b>	<p><b>Enable (Default)</b> - Adds rules to the firewall to allow link local addresses traffic on workload.</p> <p><b>Disable</b> - Does not add any rules. The Multicast traffic drops if the default policy on the agent is DENY.</p>
<b>CPU Quota Mode for enforcement process</b>	<p><b>Adjusted (Default)</b> - The CPU limit adjusts according to the number of CPUs on the system. For example, if there are 10 CPUs, set the CPU limit to 3%, the agents use only a total of 30% (measured by top).</p> <p><b>Top</b> - The CPU limit value matches the top view on average. For example, if you set the CPU limit to 3% and there are 10 CPUs in the system, the CPU usage is 3%. It is a fairly restrictive mode, use it only when necessary.</p> <p><b>Disable</b> - Disable the CPU limit feature. The agent uses CPU resources that used in the operating system.</p> <p>For more information, see <a href="#">Secure Workload Data Sheet</a>.</p>
<b>CPU Quota Limit (%)</b>	Specify the actual limit in percentage of the system processing power.
<b>Memory Quota Limit (MB)</b>	Specify the memory limit (in MB) for processes. If the process hits this limit, it restarts.
<b>Windows Enforcement Mode</b>	<p>On Windows workloads, agents can enforce network policies using:</p> <ul style="list-style-type: none"> <li>• <b>WFP</b> - Windows Filtering Platform (by directly programming WFP filters in the Windows Filter Engine). See <a href="#">Agent Enforcement on the Windows Platform in WFP Mode, on page 33</a>.</li> <li>• <b>WAF (Default)</b> - Windows Advanced Firewall. See <a href="#">Agent Enforcement on the Windows Platform in WAF mode, on page 30</a>.</li> </ul>

Table 8: Flow Visibility Config

Field	Description
<b>Flow Analysis Fidelity</b>	<b>Conversations</b> - Enable conversation mode on all sensors. <b>Detailed</b> (Default) - Enable detailed mode on all sensors.
<b>Data Plane</b>	<b>Enable</b> (Default) - Enable the agent to send reports to the cluster. <b>Disable</b> - Disable the agent's reports.
<b>Auto-Upgrade</b>	<b>Enable</b> (Default) - Automatically upgrade the agent when a new package is available. <b>Disable</b> - Do not automatically upgrade the agent.
<b>PID Lookup</b>	<b>Enable</b> - Enable process ID lookups on the agent. When enabled, the agent makes best-effort attempts to associate network flows with running processes in the workload. This operation is expensive, therefore the agent throttles the number of operations in each export cycle to keep the CPU overhead under control. It is possible that some flows are not associated with any processes even when you enable the config. <b>Disable</b> (Default) - Do not enable process ID lookups on the agent.
<b>CPU Quota Mode</b>	<b>Adjusted</b> (Default) - Adjust the CPU limit according to the number of CPUs on the system. For example, if there are 10 CPUs in the system, set the CPU limit to 3%. Choose this mode to allow the agent to use a total of 30% (measured by top). <b>Top</b> - The CPU limit value matches the top view on average. For example, set the CPU limit to 3% for the 10 CPUs in the system, the CPU usage is only 3%. It is a fairly restrictive mode and uses it only when necessary. <b>Disable</b> - Disable the CPU limit feature. The agent uses CPU resources that are used in the operating system.
<b>CPU Quota Limit (%)</b>	Specify the actual limit in percentage of the system processing power that the agent can use.
<b>Memory Quota Limit (MB)</b>	Specify the memory limit in MB that the process allows to use. If the process hits this limit, the process restarts.
<b>Cleanup period (days)</b>	<b>Enable</b> - Enable automated cleanup on the agent. Enter the number of days after which remove the inactive agent. <b>Disable</b> (Default) - Do not enable automated cleanup on the agent.
<b>Flow Analysis Fidelity</b>	<b>Conversations</b> - Enable conversation mode on all agents. <b>Detailed</b> (Default) - Enable detailed mode on all agents.



Field	Description
<b>Flows Disk Quota (MB)</b>	<p>Enter the maximum size limit (in MB) for storing the flow data.</p> <p>If the Flows Disk Quota field is:</p> <ul style="list-style-type: none"> <li>• 0: The agents do not store offline flows locally.</li> <li>• Blank: Enable the Flows Time Window field. After you enter the duration in the Flows Time Window, the Flows Disk Quota field automatically sets the value to 16 GB.</li> </ul> <p>You can either choose the Flows Disk Quota or the Flows Time Window option for flow log buffering in case of connectivity break between the agent and the cluster.</p> <p>For example, if you have set the Flows Time Window as one hour and the agent is unable to communicate with the cluster, the agent stores flow data for the last hour. Any flow data locally stored on the workload beyond the last hour is overridden by newer logs.</p>
<b>Flows Time Window (Hours)</b>	<p>This field is displayed only when there is no value that is entered in the Flows Disk Quota field.</p> <p>Enter the duration, in hours, for the agents to capture the flows and store them locally.</p> <ul style="list-style-type: none"> <li>• After the connectivity to the agents is restored, the agents send the live flow data.</li> <li>• While sending the live flow data, the agents also initiate to upload the buffered telemetry data. The telemetry data is sent in small packets at regular intervals.</li> <li>• Depending on the size of the buffered telemetry data and transmission transfer speed, it takes multiple intervals to send all the buffered data.</li> <li>• The agents progressively delete the locally stored flow data.</li> </ul> <p>Remove the outdated flow data that is stored locally after it reaches the configured size or time limit.</p>

Figure 6: Flow Visibility

Enforcement

Enforcement  
 Enable  Disable (Default)

Windows Enforcement Mode  
 WAF  WFP (Default)

Preserve Rules  
 Enable  Disable (Default)

Allow Broadcast  
 Enable (Default)  Disable

Allow Multicast  
 Enable (Default)  Disable

Allow Link Local Addresses  
 Enable (Default)  Disable

CPU Quota Mode  
 Disable  Adjusted (Default)  Top

CPU Quota Limit (%)

Memory Quota Limit (MB)

Figure 7:

**Flow Visibility**

Flow Analysis Fidelity

Conversations (Default)  Detailed

Data Plane

Enable (Default)  Disable

Auto-Upgrade

Enable (Default)  Disable

PID Lookup

Enable  Disable (Default)

Service Protection ⓘ

Enable  Disable (Default)

CPU Quota Mode

Disable  Adjusted (Default)  Top

CPU Quota Limit (%)

Memory Quota Limit (MB)

Cleanup Period (days) ⓘ

Flows Disk Quota (MB) ⓘ

Flows Time Window (h) ⓘ

Table 9: Process Visibility and Forensics Config

Field	Description
<b>Forensics</b>	<p><b>Enable</b> - Enable forensics on the agent. This feature consumes extra CPU cycles that are specified in the CPU limit below. For example, if the CPU limit is 3% and you enable this feature, the agent uses up to 6% in total.</p> <p><b>Disable (Default)</b> - Disable forensics on the agent.</p>
<b>Meltdown Exploit Detection</b>	<p><b>Enable</b> - Enable Forensics and Meltdown exploit detection on the agent. For more information, see Side Channel in the <a href="#">Compatibility</a>.</p> <p><b>Disable (Default)</b> - Disable Meltdown exploit detection on the agent.</p>

Field	Description
<b>CPU Quota Mode</b>	<p><b>Adjusted</b> (Default) - Adjust the CPU limit according to the number of CPUs on the system. For example, set the CPU limit to 3% with 10 CPUs in the system. Choose this mode to use a total of 30% (measured by top).</p> <p><b>Top</b> - The CPU limit value matches the top view on average. For example, set the CPU limit to 3% with 10 CPUs in the system, the CPU usage remains at 3%. Use this restrictive mode only if necessary.</p> <p><b>Disable</b> - Disable the CPU limit feature, the agent uses CPU resources permissible by the operating system.</p>
<b>CPU Quota Limit (%)</b>	Specify the actual limit, in percentage, of the system processing power the agent can use.
<b>Memory Quota Limit (MB)</b>	Specify the memory limit (in MB). If the storage limit goes beyond the specified limit, the process restarts.

**Step 6** Click **Save**

**What to do next**

Associate this profile with an agent config intent. For more information, see [Creating an Agent Config Intent, on page 61](#).

**Creating an Agent Config Intent****Before you begin**

- See [Requirements and Prerequisites for Configuring Software Agents, on page 52](#).
- Create an agent config profile. See [Create an Agent Configuration Profile, on page 54](#).

**Step 1** In the navigation bar on the left, click **Manage > Agents**.

**Step 2** Click the **Configure** tab.

**Step 3** Click the **Create Intent** button next to the **Agent Config Intent** heading.

**Step 4** Enter the appropriate values in the fields listed in the table below:

Field	Description
<b>Profile</b> (required)	Enter the name of an existing profile and select it from the dropdown menu.
<b>Filter</b> (required)	Enter the name of an existing filter or scope or select <i>Create new filter</i> from the dropdown menu. See <a href="#">Filters</a> for more information on creating filters.

**Step 5** Click **Save**.

**Figure 8: Agent Config Intents**

## Agent Config Intents

Apply profile  to filter

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

## Creating a Remote VRF configuration for agents

This is the recommended way to assign VRFs for Secure Workload software agents. Using this configuration, Secure Workload appliance assigns VRFs to software sensors based on the source IP address and source port seen for those agent on connections to Secure Workload appliance.

- Step 1** In the navigation bar on the left, click **Manage > Agents**.
- Step 2** Click the **Configure** tab.
- Step 3** Click the **Create Config** button next to the **Agent Remote VRF Configurations** heading.
- Step 4** Enter the appropriate values in the fields and click **Save**.

*Figure 9: Remote VRF configuration*

### Agent Remote VRF Configurations

Apply VRF  
 ▼

Source Subnet

Source Port Start

Source Port End

## Create an Interface Configuration Intent

We recommend assigning virtual routing and forwarding (VRFs) to agents in using Remote VRF configuration settings. In rare cases, when agent hosts have multiple interfaces that must be assigned to different VRFs, you can choose to assign them VRFs using Interface Configuration Intents.

- Step 1** Navigate to **Manage > Agents**.
- Step 2** Click the **Configure** tab.
- Step 3** Click the **Create Intent** button next to the **Interface Config Intent** heading.
- Step 4** Enter the appropriate values in the fields listed in the table:

Field	Description
<b>VRF</b>	Choose a VRF from the drop-down list (required).

Field	Description
<b>Filter</b>	Enter the name of an existing filter or scope or choose <i>Create a new filter</i> from the drop-down list (required). For more information, see <a href="#">Filters</a> .

**Step 5** Click **Save**.

**Figure 10: Interface Configuration Intents**

Interface Config Intents

Apply VRF Default to filter

Save Cancel

No intents found

Agent Remote VRF Configuration Create Config

No configs found

Everything  
Filter  
Test  
Default  
Unknown  
Tetration  
Tetration:Campus  
Tetration:Internet

Create new filter  
5 of 42 matching scopes shown

**Note** When you delete an interface with a higher priority config intent, the agents do not fall back to the default catch all intent.

## View Detailed Agent Status in the Workload Profile

**Step 1** Follow the steps above to check Agent status.

**Step 2** On the Enforcement Agents page, click **Agent OS Distribution**. Select an operating system and click filter image on the top-right corner of the box.

**Step 3** On the Software Agent List page, agents with selected operating system Distribution is listed.

**Step 4** Click on **Agent** for the agent details, and click IP address. On the Workload Profile page, you can view details of the Host Profile, Agent Profile and agent specific details, such as Bandwidth, Long-lived Processes, Packages, Process Snapshot, Configuration, Interfaces, Stats, Policies, Container Policies and so on.

**Step 5**

**Step 6** Click **Config tab** to see the configuration on the endhost.

**Step 7** Click **Policies** tab to see the enforced policies on the endhost.

**Figure 11: Workload Profile - Config**

**Figure 12: Workload Profile - Policies**

Priority	Packets Tl	Bytes Tl	Actions Tl	Direction Tl	Family Tl	Proto Tl	Src Inventory Tl	Src Ports Tl	Dest Inventory Tl	Dest Ports Tl
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



# Rehoming of Agents

Rehoming of agents is the method to move users from On-premises to SaaS or SaaS to On-premises.

## User Roles

- Site Administrator
- Customer Support Representative

You can migrate to or from a SaaS environment, especially, when you move from SaaS to On-premises, you must work with an internal support team.

## Workflow

- Enter the Activation Key, Sensor virtual IP, and Sensor certificate authority (CA) and [Enable Rehoming, on page 65](#).
- [Select Agents to Rehome, on page 67](#).
- [Disable Rehoming, on page 67](#).



**Note** At any given time, you can move an agent to only one destination. We recommend that you Disable Agent Rehoming after you move the agent.

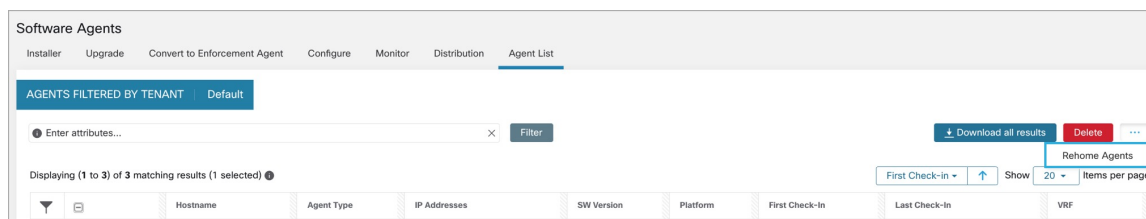
## Enable Rehoming

**Step 1** In the left navigation menu, click **Manage > Workloads > Agents**.

**Step 2** Click the **Agent List** tab.

**Step 3** Click the menu icon and select **Rehome Agents**.

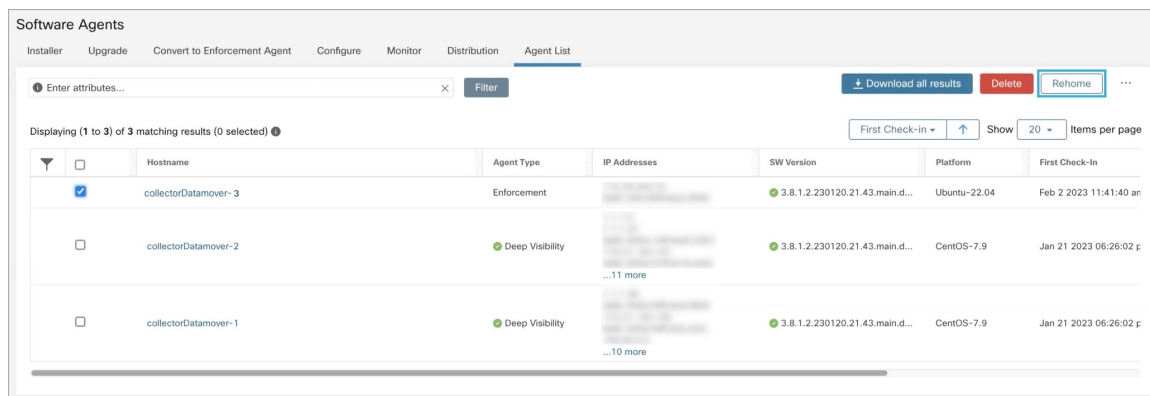
**Figure 13: Rehome Agents**



**Step 4** On the **Agent Rehoming** window, fill in the following details:

Field	Description
<b>Destination Scope Activation Key</b>	<ol style="list-style-type: none"> <li>Navigate to <b>Manage &gt; Workloads &gt; Agents</b>.</li> <li>Click the <b>Installer</b> tab.</li> <li>Select <b>Manual install using classic packaged installers</b>.</li> <li>Click <b>Next</b>.</li> <li>Click <b>Agent Activation Key</b>.</li> <li>Copy the <b>Key</b> value and paste it into the <b>Destination Scope Activation Key</b> field.</li> </ol>
<b>Destination Sensor VIP</b>	<ol style="list-style-type: none"> <li>Navigate to <b>Platforms &gt; Cluster Configuration</b>.</li> <li>Copy the <b>Sensor VIP</b> and paste it into the <b>Destination Sensor VIP</b> field.</li> </ol>
<b>HTTPS proxy</b>	Enter a proxy domain or address if the agent needs to use a proxy for outbound communication.
<b>Destination Sensor CA Cert</b>	<ol style="list-style-type: none"> <li>Navigate to <b>Platforms &gt; Cluster Configuration</b>.</li> <li>Click <b>Download Sensor CA Cert</b>.</li> </ol>

Figure 14: Enable Agent Rehoming

**Step 5** Click **Enable Agent Rehoming**.

The configuration is saved. The Rehome button appears at the top right.

## Select Agents to Rehome

**Step 1** Select an agent.

**Step 2** Click **Rehome**.

*Figure 15: Select Agents to Rehome*

Software Agents

Installer Upgrade Convert to Enforcement Agent Configure Monitor Distribution Agent List

Enter attributes... Filter Download all results Delete Rehome ...

Displaying (1 to 3) of 3 matching results (0 selected)

First Check-in ↑ Show 20 Items per page

	Hostname	Agent Type	IP Addresses	SW Version	Platform	First Check-In
<input checked="" type="checkbox"/>	collectorDatamover-3	Enforcement	...	3.8.1.2.230120.21.43.main.d...	Ubuntu-22.04	Feb 2 2023 11:41:40 an
<input type="checkbox"/>	collectorDatamover-2	Deep Visibility	...11 more	3.8.1.2.230120.21.43.main.d...	CentOS-7.9	Jan 21 2023 06:26:02 p
<input type="checkbox"/>	collectorDatamover-1	Deep Visibility	...10 more	3.8.1.2.230120.21.43.main.d...	CentOS-7.9	Jan 21 2023 06:26:02 p

**Step 3** Click **Yes** to confirm.

## Disable Rehoming



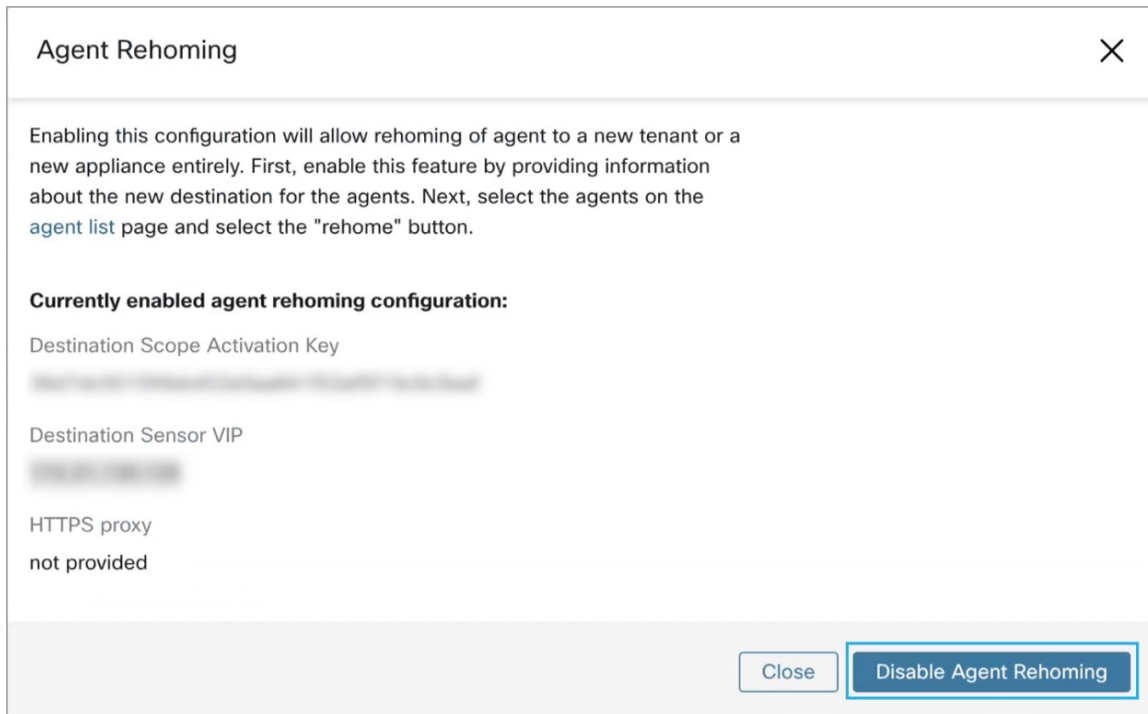
**Note**

If there are multiple users rehoming to or from SaaS, the site administrator has to move each tenant or an appliance separately. To do this, disable Rehoming to clear the settings, and then enable Rehoming for the new user.

**Step 1** Click the menu icon and choose **Rehome Agents**.

**Step 2** On the **Agent Rehoming window**, click **Disable Agent Rehoming**.

Figure 16: Disable Agent Rehoming



## Host IP Address Change when Enforcement is Enabled

Changing the IP address on hosts when enforcement is enabled may have an impact if the host IP is seen in the host firewall rules and catch all is set to deny. In this scenario, the following steps are recommended to change the host IP address:

- Step 1** On the Secure Workload UI, create a new Agent Config Profile with enforcement disabled.
- Step 2** Create Intent with list of hosts that need IP address change with their old and new IP address.
- Step 3** Apply the newly created Agent Config Profile to the Intent and save the Intent.
- Step 4** These selected hosts should have enforcement disabled.
- Step 5** Change the IP address on these hosts.
- Step 6** On the Secure Workload UI, update the filters in the scope with the new IP address of these hosts.
- Step 7** Verify the IP address change from Agent Workload Profile page “Interfaces” tab. In the “Policies” tab, make sure policies are generated with new IP address.
- Step 8** Remove the Intent/Profile created above.
- Step 9** If the original Agent Config Profile for the scope had enforcement disabled, then enable enforcement.

# Upgrading Software Agents

## Upgrade Agents from UI

Agents can be upgraded using Agent Config Intent workflow as described here - [Software Agent Config](#). While configuring an agent config profile, there is an **Auto Upgrade** option which can be enabled or disabled. If the option is enabled, the agents matching inventory filter criteria are automatically upgraded to the latest available version.

To use software agent config intent workflow to configure software agent upgrade:

**Step 1** Create an inventory filter on the **Inventory Filters** page. For more information, see [Filters](#).

**Figure 17: Inventory Filter**

+ 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](#)

Cancel Previous Next

**Step 2** Create an Agent Config profile for the agents selected by the inventory filter. Optionally, you can enable the **Auto Upgrade** option to automatically upgrade the selected agents.

Figure 18: Agent Config

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

[View Deleted Agent Config Profiles](#) 

**Step 3** Create an agent config intent to apply the config profile to the agents selected using inventory filter. If the auto upgrade option is enabled, the selected agents are automatically upgraded.

It normally takes up to 30 minutes to upgrade an agent after an agent profile is applied to them.

Figure 19: Agent Config Intent

Agent Config Intents

Apply profile  to filter

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

**Note** Auto Upgrade setting in the default agent profile applies to ERSPAN.

## Manual Agent Upgrade

The following section explains how to manually upgrade agents without using the Sensor Config intent workflow.

**Step 1** In the left navigation pane, click **Manage > Workloads > Agents**.

**Step 2** Click the **Upgrade** tab.

Deep visibility and enforcement agents are displayed and for each agent only newer versions to which it is upgradable are listed. By default, the latest version is selected.

**Step 3** To filter specific agents, enter your search query in the filter box. For example, enter Platform = CentOS-7.6.

**Step 4** Select the agents to be upgraded to the selected version and click **Upgrade**.

**Note** Under normal circumstances, allowing the agent to automatically upgrade is strongly recommended and is the only supported upgrade method. If you want to control the upgrade by manually downloading the latest version and directly deploying it to the agents which are running on workloads, ensure that you follow the safety precautions.

## Upgrade Behaviour of Kubernetes/Openshift Agent

Agents installed on Kubernetes/Openshift nodes using the daemonset installer script are capable of self-upgrade. The upgrade process is controlled by either the auto-upgrade option or by manually triggering an upgrade for any node in the Kubernetes/Openshift cluster. The mechanism of the upgrade in this environment is to upgrade the Docker image in the daemonset specification, which means that an upgrade of one agent affects all agents covered by the daemonset, as explained in the next paragraph.

When a Daemonset Pod specification changes, Kubernetes/Openshift will trigger a graceful shutdown, fetch the new docker image(s) and start the Secure Workload agent pods on ALL nodes in the Kubernetes/Openshift cluster. This will cause agents to be upgraded on other nodes, even if the policy to allow upgrades is applicable only to a subset of the nodes in the cluster.

If auto-upgrade is disabled for all nodes, manual upgrade is possible by downloading a new installer script and re-running the install. The installation script auto-detects the case of new installation vs upgrading an existing installation and will work to manually upgrade the daemonset pods when it detects an installation is already in place.

## Removing Software Agents

### Remove a Deep Visibility or Enforcement Linux Agent

**RPM based installation:**

1. Run command: `rpm -e tet-sensor`

Manually delete the agent from UI on **Software Agent** page.

**Ubuntu .deb based installation:**

Fresh installation of Ubuntu agents now uses the native .deb format.

1. Run command: `dpkg --purge tet-sensor`

Manually delete the agent from UI on **Software Agent** page.



**Note**

- During the agent operations, it is possible that some kernel modules will be loaded automatically by the kernel. For example, if enforcement is enabled in Linux, Netfilter modules might be loaded. Agents do not have a list of modules loaded by kernel. Therefore, during agent uninstallation, it cannot possibly unload the kernel modules.
- If enforcement agent applied a policy to the system firewall, uninstalling agent clears the applied policy and opens the system firewall.

### Removing a Deep Visibility/Enforcement Windows Agent

There are two options to uninstall Secure Workload agents:

- Step 1** Go to Control Panel / Programs / Programs And Features, and uninstall **Cisco Secure Workload Agent** (Cisco Tetration Agent).
- Step 2** Alternatively, run the shortcut **Uninstall.lnk** within **'C:\Program Files\Cisco Tetration'**
- Step 3** If enforcement agent applied a policy to the system firewall, uninstalling agent clears the applied policy, and opens the system firewall.

Manually delete the agent from UI on the **Software Agent** page .

**Note** By default, the **cleanup period** is turned off.



- Note**
- If Npcap has been installed during agent installation, it will also get uninstalled.
  - By default log files, config files and certs will not get removed during uninstall. If you'd like to remove them, run the shortcut **UninstallAll.lnk** in same folder.
- 

## Remove a Deep Visibility or Enforcement AIX Agent

---

Run command: `'installp -u tet-sensor'`.

Manually delete the agent from UI on the **Software Agent** page .

- Note**
- The Deep Visibility Agent is controlled by System Resource Controller as tet-sensor. It is possible to start, stop, restart, and remove it. The service is made persistent with inittab as tet-sen-engine.
  - The Enforcement Agent is controlled by System Resource Controller as tet-enforcer. It is possible to start, stop, restart, and remove it. The service is made persistent with inittab as tet-enf-engine.
  - During the agent operations, it is possible that some kernel modules will be loaded automatically by the kernel. For example, if enforcement is enabled in AIX, ipfilter modules are loaded. Agents do not have a list of modules loaded by kernel. Therefore, during agent uninstallation, it cannot possibly unloaded the kernel modules.
  - If enforcement agent applied a policy to the system firewall, uninstalling agent clears the applied policy and opens the system firewall.
- 

## Remove Universal Linux Agent

---

**Step 1** Run the uninstall script: `'/usr/local/tet-light/uninstall.sh'`

**Step 2** Delete the agent from UI on the **Software Agent** page

---

## Remove Universal Windows Agent

---

**Step 1** Run the uninstall script: `'C:\Program Files\Cisco Tetration\Lightweight Sensor\uninstall.cmd'`

**Step 2** Delete the agent from UI on the **Software Agent** page

---

## Remove an Enforcement Kubernetes or OpenShift Agent

---

- Step 1** Locate the original installer script or download a new script from the Secure Workload UI.
- Step 2** Run the uninstall option: **install.sh --uninstall**. The same considerations apply as during the install.
- Only supported on Linux x86\_64 architectures.
  - Either `~/.kube/config` contains an admin credentials user or use the `--kubeconfig` option to point to the `kubectl` admin credentials file.
- Step 3** Delete the agents for all the Kubernetes nodes from UI on the **Software Agent** page
- 

## Data collected and exported by workload agents

This section describes the main components of a software agent, how it is registered with backend services, what data are collected and exported to the cluster for analytical purposes.

### Registration

After the agent has been successfully installed onto the system, it needs to register with the backend services to obtain a valid unique identifier. The following information is sent in the registration request:

- Hostname
- BIOS-UUID
- Platform information (such as CentOS-6.5)
- Self-generated client certificate (generated with `openssl` command)
- Agent type (visibility or enforcement.)

If the agent fails to obtain a valid id from the server, it will keep retrying until it gets one. It is very important that the agent is registered, otherwise all the subsequent communication with other services (such as collectors) will be rejected.

### Agent upgrade

Periodically (around 30 minutes), the agent sends a message to backend service to report its current version. The backend service uses the agent's id and its current version to decide whether a new software package is available for the agent. The following information is sent:

- Agent's id (obtained after successful registration)
- Current agent's version

## Config server

Agents export the following information to the configured config server:

- Hostname
- Agent's id (obtained after successful registration)
- List of interfaces, each includes:
  1. Interface's name
  2. IP family (IPv4 or IPv6)
  3. IP addresses
  4. Netmask
  5. Mac addresses
  6. Interface's index

As soon as any interface property changes (such as an IP address of an existing interface changes, or a new interface comes up), this list is refreshed and reported to the config server.

## Network Flow

Network flow information is the summarization of all packets flowing through the system. There are two modes of capturing flow information: Detailed and Conversation. By default, the Detailed mode of capture is used. The captured flows are exported to a collector and the exported information includes:

- Flow identifier: Uniquely identify the network flow. It includes the general information such as: IP protocol, source and destination IP, and layer 4 ports.
- IP Information: Contains information that is seen in the IP header, such as: TTL, IP flags, Packet ID, IP options, and Fragmentation flags.
- TCP Information: Contains information that is seen in the TCP header, such as: sequence number, Ack number, TCP options, Rcvd windows size.
- Flow Information: Statistics of the flow (such as total packets, total bytes, TCP flags statistics, packet length statistics, and socket statistics), interface index from which the flow was observed, start time and end time of flow.

In Conversation mode, the agent exports only TCP flows that are bidirectional in nature along with other connectionless flows. Conversation mode is supported for Windows, AIX, and Linux platforms. For more information on Conversation mode, see [Conversation Mode](#).



---

**Note**

- In K8s environment, correlation of Pod or Host flows are not done in Conversation mode.
  - In either of the modes, agents do not export the following flows:
    - ARP/RARP conversations
    - Agent's flows to collectors
-

## Machine information

Machine info describes all the processes running on the host. In addition, it contains network information that is associated with the processes and the command used to launch the processes. Machine info is exported every minute and includes the following information:

- Process ID
- User ID: owner of the process
- Parent Process ID
- Command string used to launch the process
- Socket information: protocol (such as UDP or TCP), address type: IPv4 or IPv6, source and destination IP, source and destination port, TCP state, process's start and end time, path to process binary
- Forensic information: for more information, see the section [Compatibility](#).

## Agent statistics

Agent keeps track of various statistics, including system's statistics and its own, such as:

- Agent's start time and uptime
- Agent's run time in user mode and kernel mode
- Number of packets received and dropped
- Number of successful and failed SSL connections
- Total flow packets and bytes
- Total exported flows and packets to collectors
- Agent's memory and CPU usage

## Enforcement Alerts

There are three types of enforcement alerts:

- Agent Reachability

This alert detects when the agent is not reachable. This alert triggers if the agent has not communicated with the Secure Workload cluster for more than the configured number of seconds.

### 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

[Low](#) [Medium](#) [High](#) [Critical](#) [Immediate Action](#)

Hide Advanced Settings ^

Individual Alerts

[Enable](#) [Disable](#)

Summary Alerts

[None](#) [Hourly](#) [Daily](#)

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

- Workload Firewall

This alert triggers if enforcement is configured on a workload but the workload Firewall is detected to be off, since this condition will prevent Secure Workload Agent from enforcing traffic policies.

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

- Workload Policy

This alert triggers if the workload firewall rules are different from the Secure Workload policies applicable to this workload (the workload's "concrete policies".)

### 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

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

You can set the Severity of the alert as well as other per-type configuration parameters.

To configure enforcement alerts, see [Configure Alerts](#).

Figure 20: Configuring Enforcement alerts

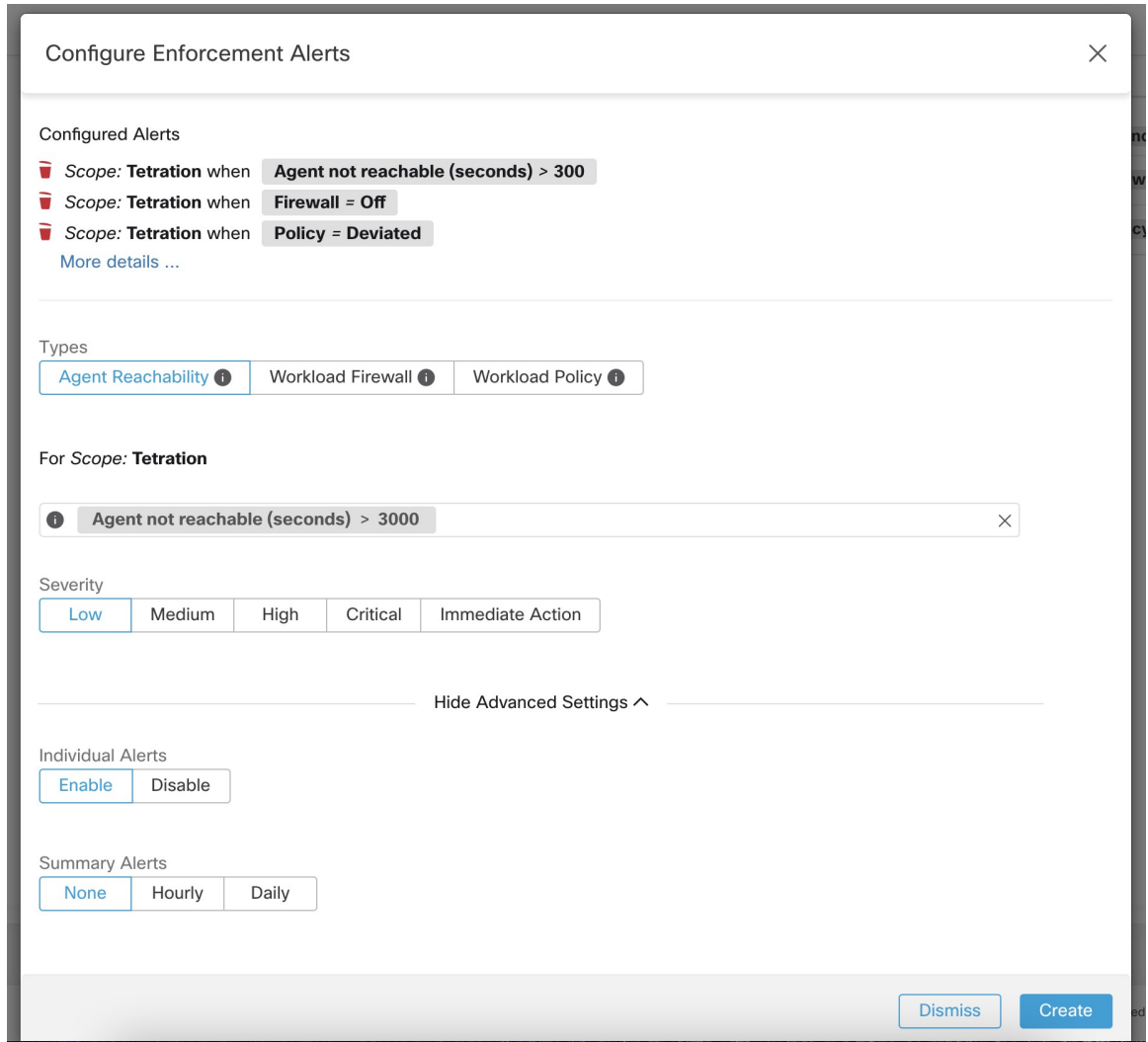


Figure 21: Viewing configured Enforcement Alerts on the alerts configuration page

### Alerts Trigger Rules

× Filter Alerts

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



# Enforcement UI Alerts Details

Figure 22: Enforcement alert details

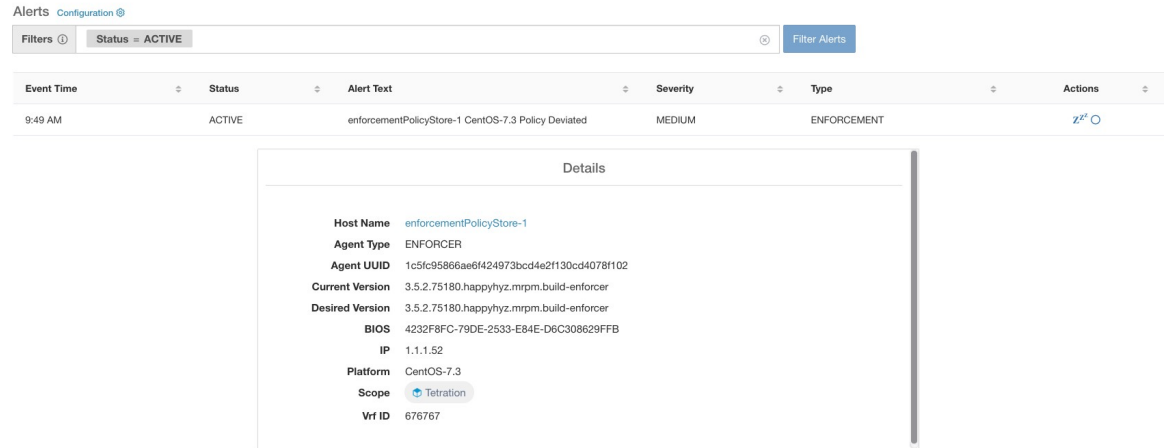
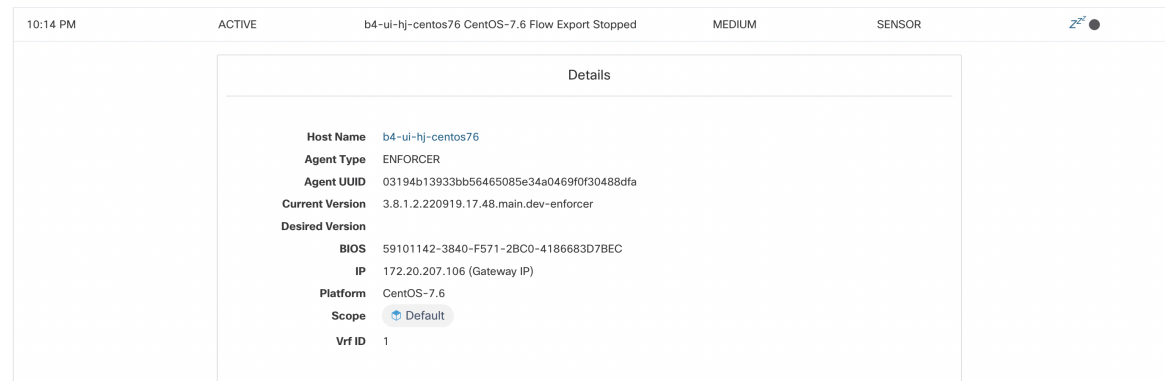


Figure 23: Enforcement alert details when proxy is enabled on the host



## Enforcement Alert Details

See [Common Alert Structure](#) for general alert structure and information about fields. The `alert_details` field is structured and contains the following subfields for enforcement alerts

Field	Alert Type	Format	Explanation
AgentType	<i>all</i>	string	“ENFORCER” or “SENSOR” depending on the installed type
HostName	<i>all</i>	string	Host name on which the agent is deployed
IP	<i>all</i>	string	IP address of the node/gateway

## Example of alert\_details for an enforcement alert

Field	Alert Type	Format	Explanation
Bios	<i>all</i>	string	BIOS UUID of the node
Platform	<i>all</i>	string	Platform/OS information of the node
CurrentVersion	<i>all</i>	string	Software version of the agent on the node
DesiredVersion	<i>all</i>	string	Software version desired for the agent
LastConfigFetchAt	<i>all</i>	integer	Unix timestamp of when the agent last sent https request

## Example of alert\_details for an enforcement alert

```
{
  "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"
}
```

## Sensor Alerts

For general information about the mode, see [Alert Configuration Modal](#).




---

**Note** Starting Secure Workload 3.5, you can configure Sensor Alerts, using the *Alert Configuration Model*.

---

Figure 24: Configure Sensor Alerts

**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

Sensor Alert Configuration provides the ability to configure three different types of alerts, you can set the severity of the alert and types of configuration parameters.

### 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 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

Configure Sensor alerts to report when an agent fails to upgrade. This alert triggers if the agent failed to upgrade to the needed version.

**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

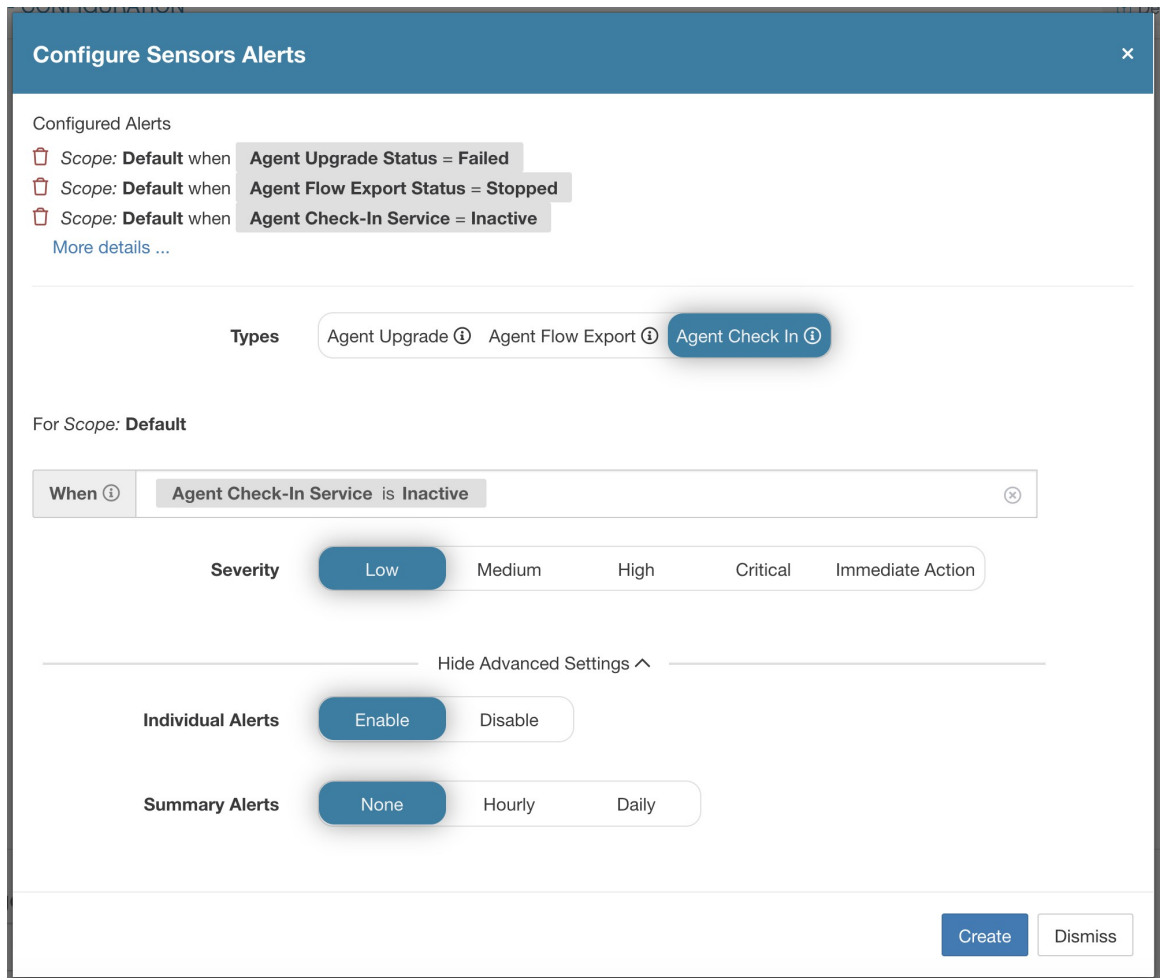
Hide Advanced Settings ^

**Individual Alerts** Enable Disable

**Summary Alerts** None Hourly Daily

**Create** **Dismiss**

Configure Sensor alerts to detect when agent flow export must stop. This alert triggers if connectivity is blocked between the agent and the cluster, therefore preventing flows and other system information from sent or delivered.



Configure sensor alerts to detect when agent check\_in times out. This alert triggers if the cluster does not received a check-in request from an agent after more than 90 minutes.

**Figure 25: Configure Sensor Alerts on Alerts Configuration**

Alerts Trigger Rules

Filters

Alert Type	Configuration	Actions
SENSORS	Scope: Default when Agent Upgrade Status = Failed	<input type="button" value="trash"/>
SENSORS	Scope: Default when Agent Flow Export Status = Stopped	<input type="button" value="trash"/>
SENSORS	Scope: Default when Agent Check-In Service = Inactive	<input type="button" value="trash"/>

## Sensor UI Alerts Details

Figure 26: Sensor Alerts

Alerts Configuration

Filters Status = ACTIVE Filter Alerts

Event Time	Status	Alert Text	Severity	Type	Actions
11:13 AM	ACTIVE	b4-ui-centos76 CentOS-7.6 Agent Inactive	MEDIUM	SENSOR	<a href="#">Z</a>

Details

**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**

**BIOS** 59101142-3840-F571-2BC0-4186683D7BEC

**IP** 172.20.207.106

**Platform** CentOS-7.6

**Scope** Default

**Vrf ID** 1

## Sensor Alert Details

For the general structure of alerts and for information about fields, see Common Alert Structure. The `alert_details` field is structured and contains the following subfields for sensor alerts

Field	Alert Type	Format	Explanation
AgentType	<i>all</i>	string	ENFORCER or SENSOR depending on the installed type
HostName	<i>all</i>	string	Host name on which the agent is deployed
IP	<i>all</i>	string	IP address of the node/gateway
Bios	<i>all</i>	string	BIOS UUID of the node
Platform	<i>all</i>	string	Platform/OS information of the node
CurrentVersion	<i>all</i>	string	Software version of the agent on the node
DesiredVersion	<i>all</i>	string	Software version desired for the agent
LastConfigFetchAt	<i>all</i>	integer	Unix timestamp of when the agent last sent HTTPS request

## Example of alert\_details for a sensor alert

```
{
  "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"
}
```