PROFITAP

SUPERVISOR

Profitap Centralized Management

USER MANUAL

Supervisor software version: v1.3.0

If you have any questions, visit our Knowledge Base:

https://kb.profitap.com/

You can also contact us through our website:

https://www.profitap.com/contact-us/

Or directly by email:

support@profitap.com

For the latest documentation and software, visit our Resource Center:

https://resources.profitap.com/

TABLE OF CONTENTS

1. Supervisor Overview

Profitap Supervisor is a centralized management platform for orchestrating physical and virtual monitoring tools from a single interface. It consists of two modules that can be used independently or together: **Network Packet Broker CM** and **Cloud TAP**.

Network Packet Broker CM allows you to organize and control all XX-Series and X2-Series Network Packet Brokers deployed inside your network architecture. It provides a comprehensive overview of the connected monitoring fabric and brings this together into a single interface. Network Packet Broker CM helps orchestrate clusters of devices all at once, instead of maintaining each device separately. By automating update and maintenance processes, it simplifies the workflow of managing your network monitoring infrastructure.

Cloud TAP gives you full visibility into Kubernetes/AWS EKS clusters and Azure VMs, enabling you to monitor, filter, and encapsulate traffic from K8s clusters with granularity from service to pod, and from GNU/Linux and Microsoft Windows virtual machines in Azure environments.

2. Supervisor Deployment and Update

2.1. Installation

2.1.1. Prerequisites

Supervisor is a containerized application, provided as a *docker* image.

In order to perform the application deployment, the following elements are necessary:

- *docker* installed and running;
- Profitap Supervisor *docker* image (provided);
- Profitap Supervisor license file (provided).

2.1.2. System Requirements

Minimum:

- Processor: Any physical x86_64 CPU with 4 threads capability and a top frequency of 2.40 GHz;
- System RAM: 4 GB;
- Available disk space: 2 GB.

Suggested:

- Processor: Any physical x86_64 CPU with 8 threads capability and a top frequency of 3 GHz;
- System RAM: 8 GB;
- Available disk space: 4 GB.

2.1.3. Installation

The installation can be performed using the following commands in order:

1. Create a directory to be used to store the supervisor configuration and license:

```
mkdir -p /home/user/supervisor-data/
```

This is only a reference path used for this documentation. If a different path is used, edit the following commands accordingly.

2. Copy the provided license file in the data directory:

```
cp SFM-010010-10.lic /home/user/supervisor-data/license.lic
```

Replace the name of the file in this command with the actual license file provided.

3. Load the provided Supervisor *docker* container (replace 'X.Y.Z' with the appropriate version number):

```
docker load -i profitap-supervisor-vX.Y.Z.tar
```

4. Run the Supervisor *docker* container, specifying the correct data directory (replace 'X.Y.Z' with the appropriate version number):

```
docker run -v /home/user/supervisor-data:/data:Z --network host --name supervisor
-d profitap-supervisor:vX.Y.Z
```

Alternatively, the container can be run using its own network with the following command instead of the one above, with '5443' used as an example port listened to on by the host machine (the Supervisor GUI will be accessible through this port):

```
docker run -v /home/user/supervisor-data:/data:Z -p 5443:443 -e
SUPERVISOR_MDNS=none --name supervisor -d profitap-supervisor:vX.Y.Z
```

At this point, the Supervisor application should be running. If you wish to verify that deployment has proceeded correctly, you can check the running containers using the following command:

```
docker ps
```

The Profitap Supervisor container should appear.

2.1.4. Scaling Up Supervisor Performance

The Supervisor container is designed to operate optimally using limited available resources. In the case of big deployments, the resources allocated by default may not be sufficient. In this case, it's possible to scale up the performance of the Supervisor application using the environment variable `SUPERVISOR_THREADS`. The default value for this parameter is 4, but it can be increased to up to 16 to take advantage of the performance offered by a CPU with a high number of cores.

The variable can be used when running the container, like in the following command:

```
docker run -v /home/user/supervisor-data:/data:Z -e SUPERVISOR_THREADS=8
--network host --name supervisor -d profitap-supervisor:vX.Y.Z
```

2.2. Installing a Custom HTTPS Certificate

1. Stop the Supervisor container:

```
docker stop supervisor
```

2. Replace the certificate (cert.pem) and key (key.pem) files in the data directory with the one you want to use.

3. Restart the Supervisor container:

```
docker run -v /home/user/supervisor-data:/data:Z --network host --name supervisor
-d profitap-supervisor:vX.Y.Z
```

If you wish to recreate a new self-signed certificate, in step 2, remove the cert.pem and key.pem files.

2.3. Installing a Custom CA Certificate

In order to use certain Supervisor functionalities, you may need to import a custom CA certificate. This is done by copying the certificate to a 'ca' directory in the Supervisor data directory.

1. Stop the Supervisor container:

```
docker stop supervisor
```

2. Create the 'ca' directory if necessary:

```
mkdir -p /home/user/supervisor-data/ca/
```

3. Copy the custom CA certificate to the 'ca' directory:

```
cp private-ca-cert.crt /home/user/supervisor-data/ca/
```

4. Restart the Supervisor container:

```
docker run -v /home/user/supervisor-data:/data:Z --network host --name supervisor
-d profitap-supervisor:vX.Y.Z
```

2.4. Update

When using the Supervisor *docker* container, the update process simply consists of shutting down the currently-running *docker* container, and starting the new updated *docker* container using the same data directory. The new instance will perform all of the necessary data migration. It is good practice to perform a backup of the Supervisor configuration before proceeding with the update (see Configuration Backup and Restore).

2.4.1. Prerequisites

In order to perform the update, the following elements are necessary:

- Currently installed Supervisor *docker* container;
- Data directory (we are using `/home/user/supervisor-data` for this example);
- Supervisor license file.

2.4.2. Update

The steps for updating Supervisor are as follows:

1. (Optional) Backup the current Supervisor configuration (see Configuration Backup and Restore).

2. Load the new *docker* container in your local registry (replace 'X.Y.Z' with the appropriate version number):

```
docker load -i profitap-supervisor-vX.Y.Z.tar
```

3. Stop previous instance using the following command:

```
docker stop supervisor
```

4. Start a new *docker* container instance (replace 'X.Y.Z' with the appropriate version number):

```
docker run -v /home/user/supervisor-data:/data:Z --network host --name supervisor
-d profitap-supervisor:vX.Y.Z
```

2.5. Access

Once deployed, Supervisor is accessible through the following ports:

- **443:** HTTPS GUI and API access;
- **80:** HTTP redirection to HTTPS GUI;
- **8080:** HTTP API access (docker container only).

The first access is possible using the following default credentials:

- **username:** `admin`
- **password:** `admin`

Note: It is strongly recommended to change the default administrator password when first accessing Supervisor.

To access the GUI, open a web browser and enter the Supervisor address in the address bar:

`https://<ip_addr>`

`<ip_addr>` being the IP address of the machine running the *docker* container.

Login, using the appropriate account credentials.

To change the default password, click the *Default Admin* link at the bottom left of the screen and enter a new password in the *Edit User* window.

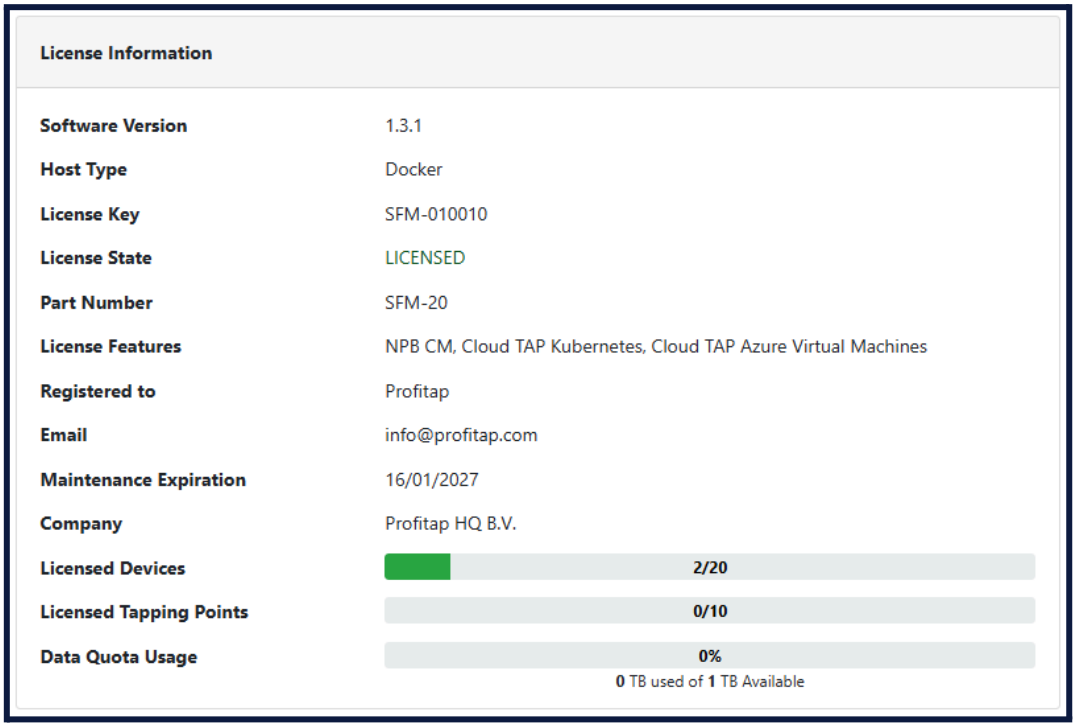
3. Supervisor Configuration

3.1. Administration

The **Administration** page can be accessed via the *Administration* menu item by users with **Administrator** role.

3.1.1. License Information

The **License Information** section of the **Setup** tab displays information about the current Supervisor license, including the licensed features, maintenance period, and licensed feature usage.



License Information section

Licensed feature usage:

- **Licensed Devices:** Number of devices registered with the NPB CM module.
- **Licensed Tapping Points:** Number of concurrent NICs being tapped with the Cloud TAP Azure VM module.
- **Data Quota Usage:** Amount of traffic mirrored with the Cloud TAP Kubernetes module.

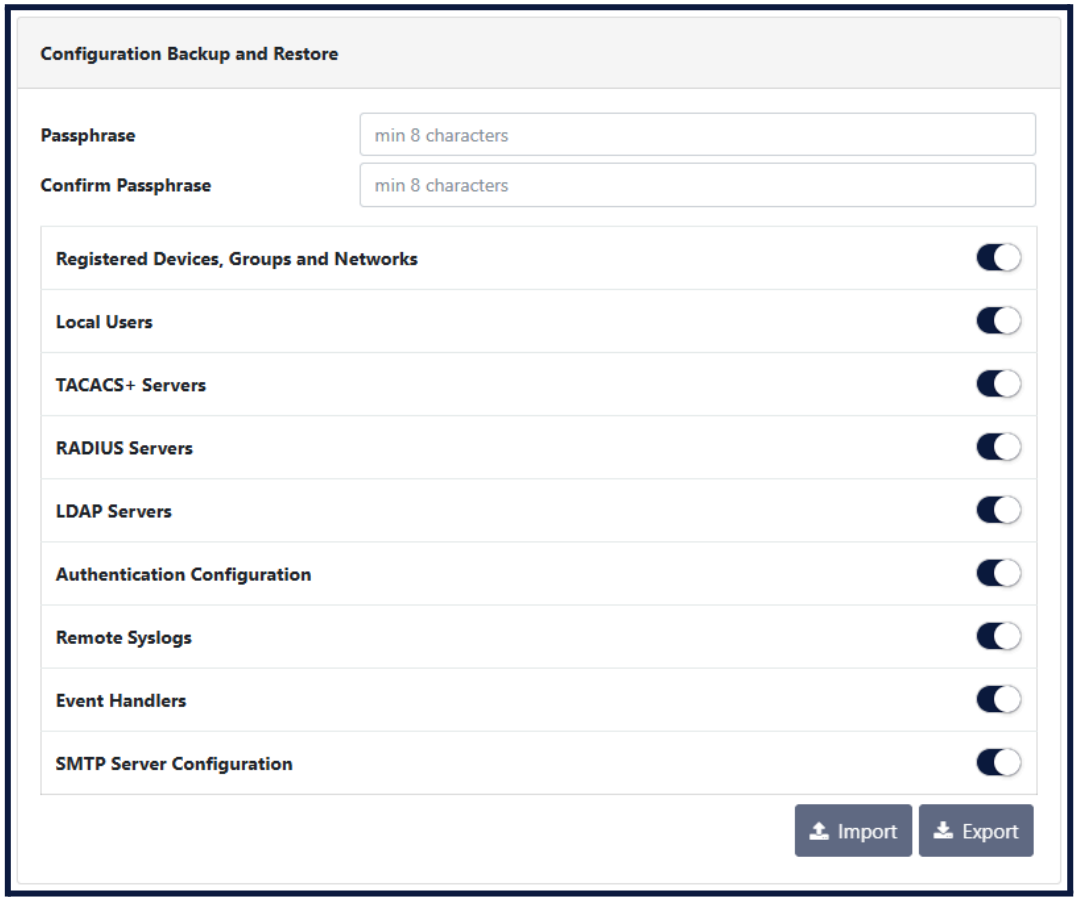
3.1.2. Configuration Backup and Restore

The **Configuration Backup and Restore** section of the **Setup** tab allows the exporting and importing of the Supervisor instance configuration. The data can be exported by inserting a passphrase, selecting the parts to be exported, and pressing the *Export* button. The system will generate an encrypted archive that can be safely stored as backup. This package can be imported back to the Supervisor instance via a similar process: insert the passphrase, select the parts of the configuration you wish to import, press the *Import* button, and select the archived configuration file.

Notes:

- The same passphrase as the one used for exporting the configuration file is required for importing it.
- The export functionality will not backup the Supervisor license.
- When restoring a backup configuration generated by Supervisor v0.9.x into Supervisor v1.0.0 and later, the external device configuration will be lost.

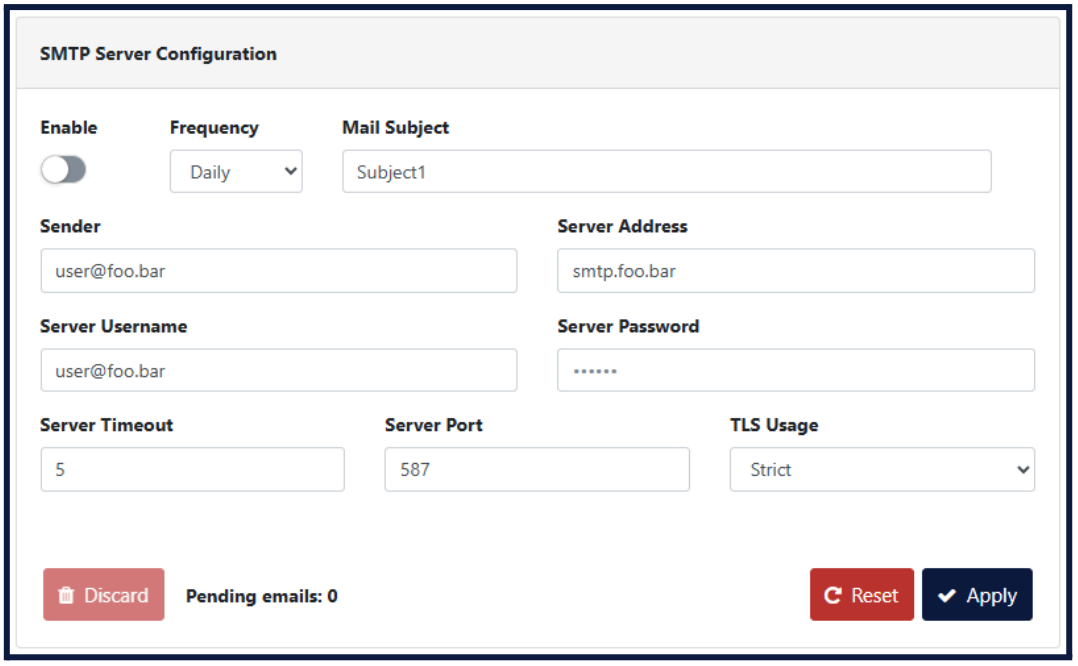
Warning: Configuration backup files generated by Supervisor v1.0.1 and earlier will not be able to be imported into Supervisor v1.1.0 and later.



Configuration Backup and Restore section

3.1.3. SMTP Server Configuration

The **SMTP Server Configuration** section of the **Setup** tab allows the configuration of settings related to the sending of alert emails (see Event Alerting).



SMTP Server Configuration section

- **Enable:** Enable or disable the sending of alert emails.
- **Frequency:** How often batch emails should be sent (hourly/daily/weekly).
- **Mail Subject:** Email subject field for batch emails.
- **Sender:** Email sender field.
- **Server Address:** Address of the SMTP server used for sending emails.
- **Server Username:** Username for authenticating with the specified SMTP server.
- **Server Password:** Password for authenticating with the specified SMTP server.
- **Server Timeout:** The maximum time the application will wait for a response from the server, in seconds. Maximum allowed: 5 seconds.
- **Server Port:** Port number for connecting to the specified SMTP server.
- **TLS Usage:**
    - **None:** TLS is not used for server communication. Data is transmitted in plain text without encryption.
    - **Lenient:** TLS is used for secure server communication, but the connection doesn't require a valid, trusted SSL/TLS certificate.
    - **Strict:** TLS is used for secure server communication, and the connection requires a valid, trusted SSL/TLS certificate.
- **Discard:** Clear the current batch of pending email alerts.
- **Reset:** Clear the current SMTP settings.
- **Apply:** Apply changes to the SMTP settings.

3.1.4. Syslog

The **Syslog** tab displays the logs of the Supervisor system. On this page, the system logs can be refreshed, downloaded, or reset. It is also possible to configure remote collectors for the system logs. This can be done by clicking the *Remote Servers* button and using the view that appears to configure the remote logging server details.

3.1.5. Support

The **Support** tab contains documentation to help you use Supervisor: a link to the Profitap Knowledge Base, accessible over the Internet, and the Supervisor manual, datasheet, and REST API documentation, which can be downloaded directly from the Supervisor instance. A section for contacting Customer Support is also present.

3.2. Authentication

The **Authentication** page can be accessed via the *Authentication* menu item by users with **Administrator** role.

3.2.1. Users

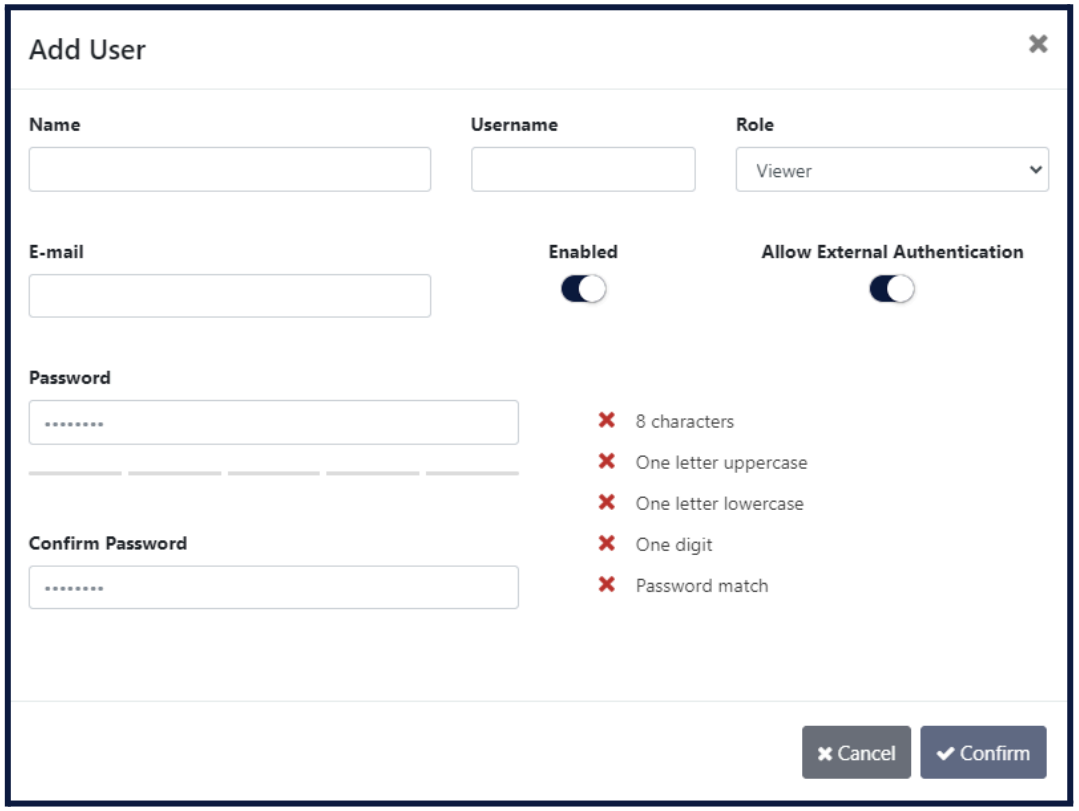
The **Users** tab allows administrators to add new users or edit existing users and their privilege levels. Depending on the selected role, the user has the following rights:

- **administrator**: full control, limitless administration and system update;
- **user**: create and set rules, aggregate and filter traffic, and port configuration;
- **viewer**: view only: settings, statistics, active rules.

The minimum requirements for the passwords are as follows:

- 8 characters;
- one letter uppercase;
- one letter lowercase;
- one digit.

The *Allow External Authentication* option allows the user's credentials to be used to log into devices on which *Shared Authentication* was enabled (see Centralized Authentication).



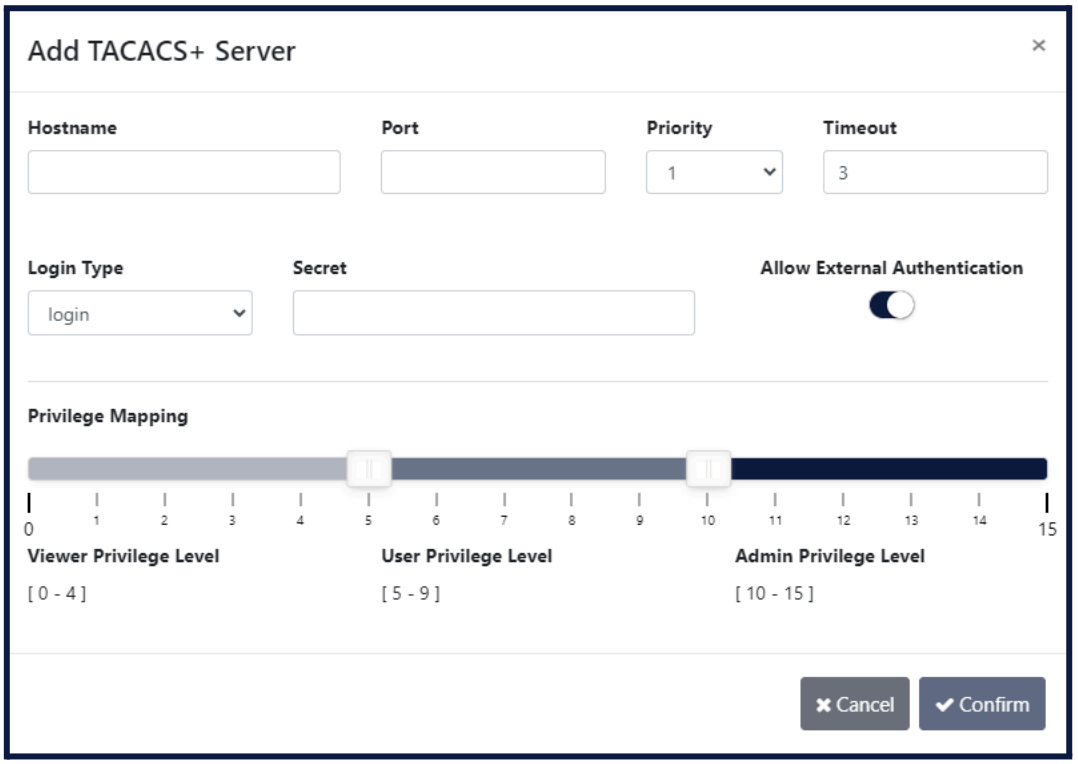
Add User window

3.2.2. TACACS+

The **TACACS+** tab allows adding one or more TACACS+ servers, and configuring the following details:

- priority (sets the order in which the servers will be taken into account, if more are added, with a lower number corresponding to a higher priority);
- login type (chap, login, pap);
- server hostname;
- port;
- secret key;
- timeout (waiting time for response from the TACACS+ server, can be set between 1 and 3 seconds);
- privilege mapping (translates the 15 privilege levels from TACACS+ into those of the viewers, users and admins; can be configured).

The *Allow External Authentication* option allows the user credentials defined on the TACACS+ server to be used to log into devices on which *Shared Authentication* was enabled (see Centralized Authentication).



Add TACACS+ Server window

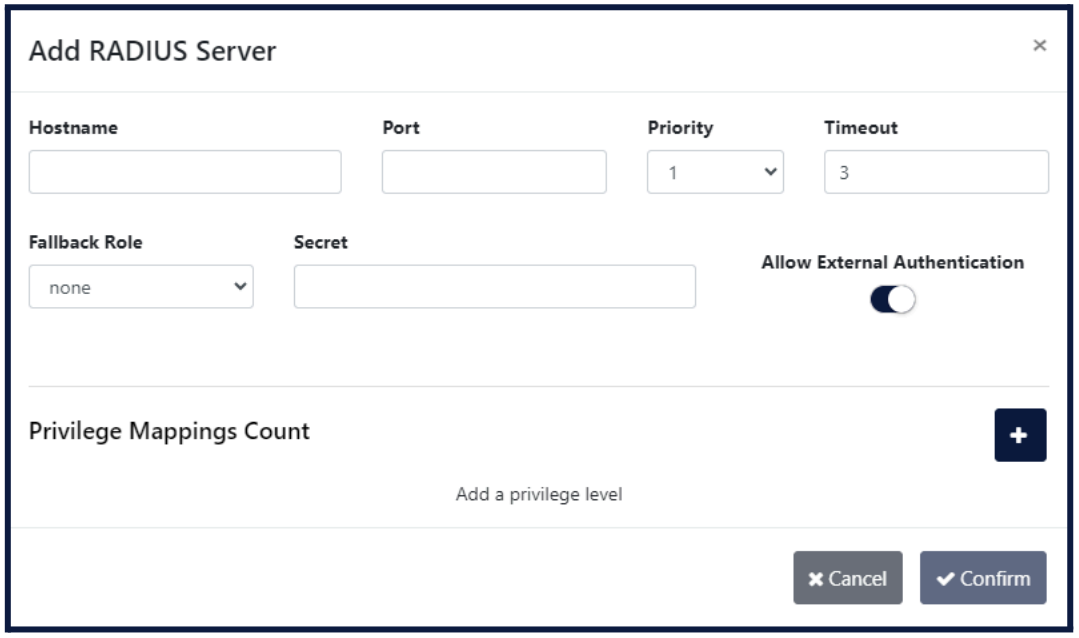
3.2.3. RADIUS

The **RADIUS** tab allows adding one or more RADIUS servers, and configuring the following details:

- priority (sets the order in which the servers will be taken into account, if more are added, with a lower number corresponding to a higher priority);
- server hostname;
- port;
- secret key;
- timeout (waiting time for response from the RADIUS server, can be set between 1 and 3 seconds);
- privilege mappings count (allows adding one or more rules for users. These rules are integer or string **type** attributes, requiring a **name** and a **value**. During authentication, the system checks if a user matches the rules. If there is a match between a user and a rule, then a **role** is applied for the user);

  Note: To add a new rule, click the + button. To apply the rule, click the ✔ button.
- fallback role (comes into place when there isn't a match between a user and a rule, with the 'none' option denying authentication access to *any* user).

The *Allow External Authentication* option allows the user credentials defined on the RADIUS server to be used to log into devices on which *Shared Authentication* was enabled (see Centralized Authentication).



Add RADIUS Server window

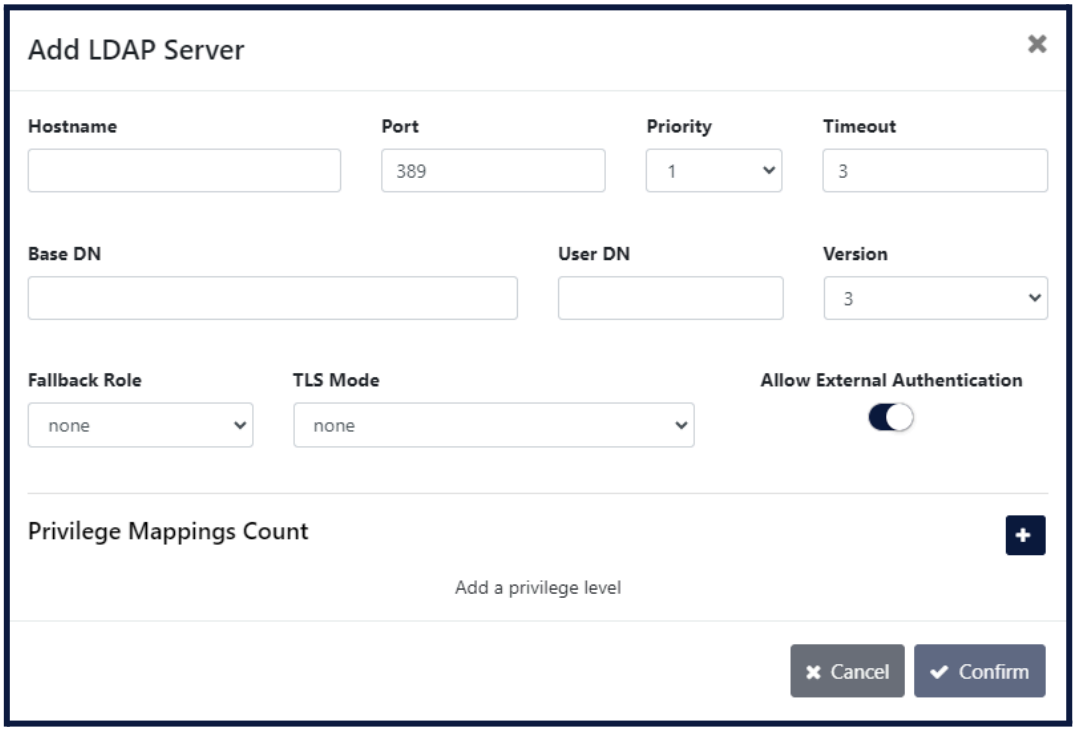
3.2.4. LDAP and LDAPS

The **LDAP** tab offers the possibility to configure one or more LDAP servers for user authentication. In order to set up the LDAP access, the following settings are required:

- server hostname or address;
- server port: (default 389 for LDAP and 636 for LDAPS);
- priority (sets the order in which the servers will be taken into account, if more are added, with a lower number corresponding to a higher priority);
- timeout (waiting time for response from the LDAP server, can be set between 1 and 3 seconds);
- base DN (base distinguished name): this is the base DN used to query the LDAP servers for its information (example: `ou=people,dc=example,dc=com`);
- user DN (user distinguished name): DN parameter used to query for the usernames. (example: `uid`);
- LDAP version: it is possible to configure both LDAP Version 2 and Version 3 servers;
- privilege mappings count (allows adding one or more rules for users. These rules are integer or string **type** attributes, requiring a **name** and a **value**. During authentication, the system checks if a user matches the rules. If there is a match between a user and a rule, then a **role** is applied for the user);

  Note: To add a new rule, click the + button. To apply the rule, click the ✔ button.
- fallback role (comes into place when there isn't a match between a user and a rule, with the 'none' option denying authentication access to *any* user);
- TLS mode: the user can select whether the server requires TLS (for LDAPS), and if they wish to enforce strict TLS session validation. Note that if this option is set to "strict", the user will likely need to import a private CA certificate into Supervisor (see Installing a custom CA certificate).

The *Allow External Authentication* option allows the user credentials defined on the LDAP server to be used to log into devices on which *Shared Authentication* was enabled (see Centralized Authentication).



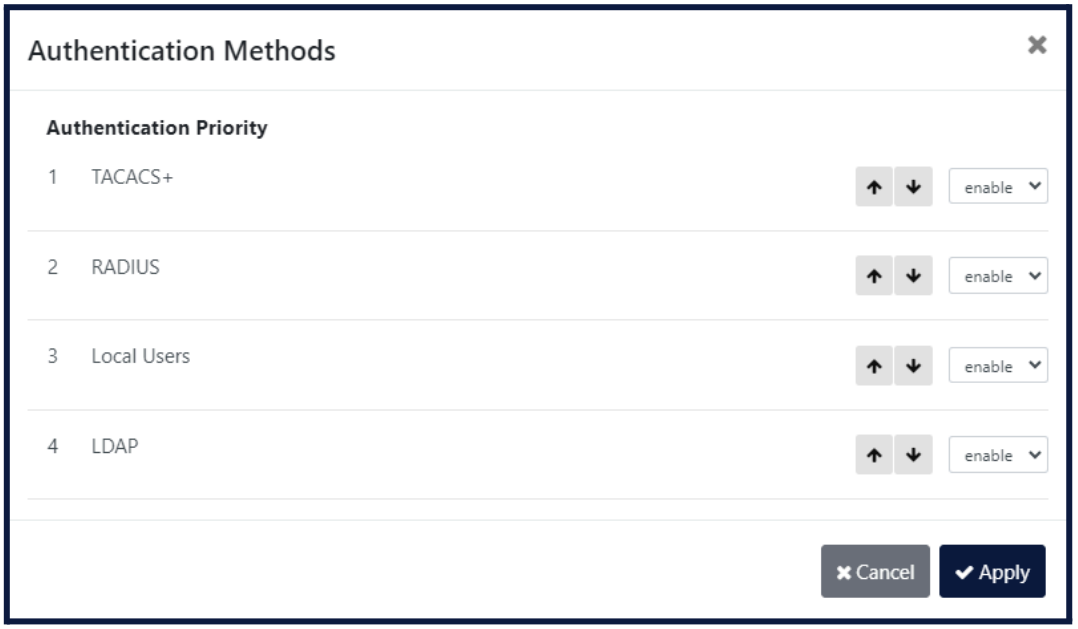
Add LDAP Server window

3.2.5. Custom Authentication Configuration

Supervisor allows users to not only define multiple authentication methods, but also to configure how the different methods are used by the system. Clicking the *Configure Authentication* button on either the *Users*, *TACACS+*, *RADIUS*, or *LDAP* page allows users to see the list of available authentication methods and change their priority and activation strategy.

For each method, one of the following strategies can be selected:

- **Enable**: The method is activated and will be used to authenticate users;
- **Disable**: The method is not active and its configuration will be ignored;
- **Restrict**: A restricted authentication method is activated only if all higher priority methods are failing access. In the case of RADIUS, LDAP, or TACACS+ methods, this means that no server is responding (or no server is programmed). If only one of the registered LDAP/RADIUS/TACACS+ servers replies with a rejection, the following restricted methods will be skipped. Note that “Local Users” are always available, meaning that any “restrict” method after that will never be activated.



Authentication Methods window

3.2.6. Centralized Authentication

Supervisor provides the ability to use credentials defined in the Supervisor itself in order to log into devices it manages. Devices on which *Shared Authentication* was enabled (see Registered Devices) will be able to use Supervisor credentials, be they Local Users, or users defined on TACACS+, LDAP, or RADIUS servers, on which *Allow External Authentication* was enabled. The Centralized Authentication follows the Supervisor's Custom Authentication Configuration.

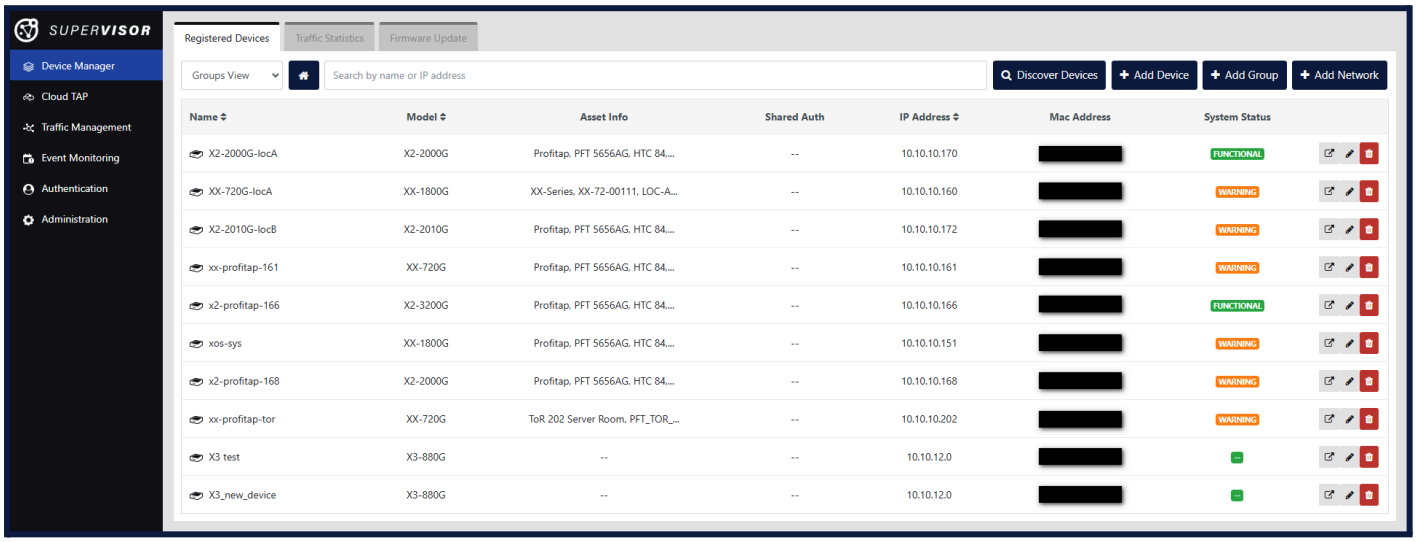
4. Network Packet Broker CM

This chapter describes the Supervisor section specific to the Network Packet Broker CM module, accessed via the *Device Manager* menu item. Devices can be managed further in the sections described in the Traffic Management and Event Monitoring chapters.

4.1. Registered Devices

4.1.1. Overview

The **Registered Devices** tab of the **Device Manager** page provides an overview of the devices managed by the Supervisor, and general information about them, such as their name, model, asset information, shared authentication status, IP address, MAC address, and system status.



List of registered devices

From this dashboard, devices, groups and networks can be added, modified, or removed.

Each device can be assigned to a **Group** and to a **Network**:

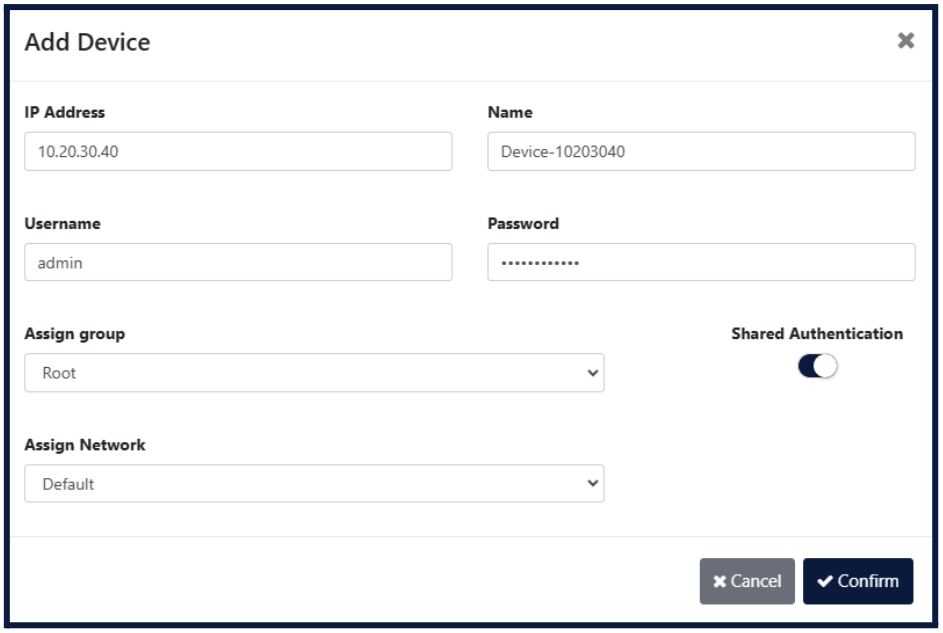
- **Groups** are used to organize devices in the Registered Devices tab (this tab), and to display traffic statistics for groups of devices in the Traffic Statistics tab.
- **Networks** are used to organize devices and virtual environments into logical networks in the Traffic Management page, with each network operating with its own set of traffic rules.

The view can be changed between *Groups View* and *Networks View* via the drop-down menu in the top left corner of the interface. The search bar can be used to filter the current view to display specific devices or groups.

Clicking on a group or network navigates to this group or network, listing the devices it contains. Clicking the *Home* button navigates back to the root. Clicking the *Open Device* button of a device opens this device's management GUI in a new tab.

To add a new device, click the *Add Device* button in the top right corner of the interface, and enter the device's information in the *Add Device* window. Select a group or network in this window to add the device

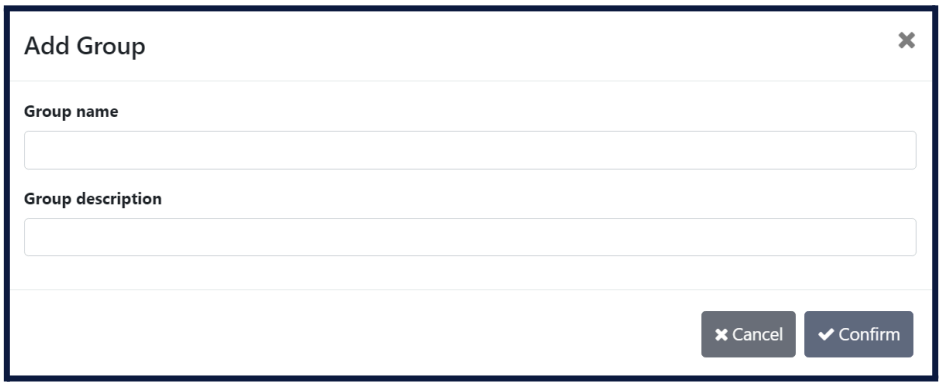
to this group or network. Enable *Shared Authentication* if you wish to enable Supervisor's centralized authentication function on this device (see Centralized Authentication). The device's information can be changed at a later time by clicking the device's *Edit* button in the list.



Add Device window

You can also add new devices via the *Discover Devices* button. The *Discover Devices* window lists devices found on the Supervisor's local network, and allows you to add them to the Supervisor.

To create a group, click the *Add Group* button, and enter the group name and description in the *Add Group* window. The group's name and description can be changed at a later time by clicking the group's *Edit* button.



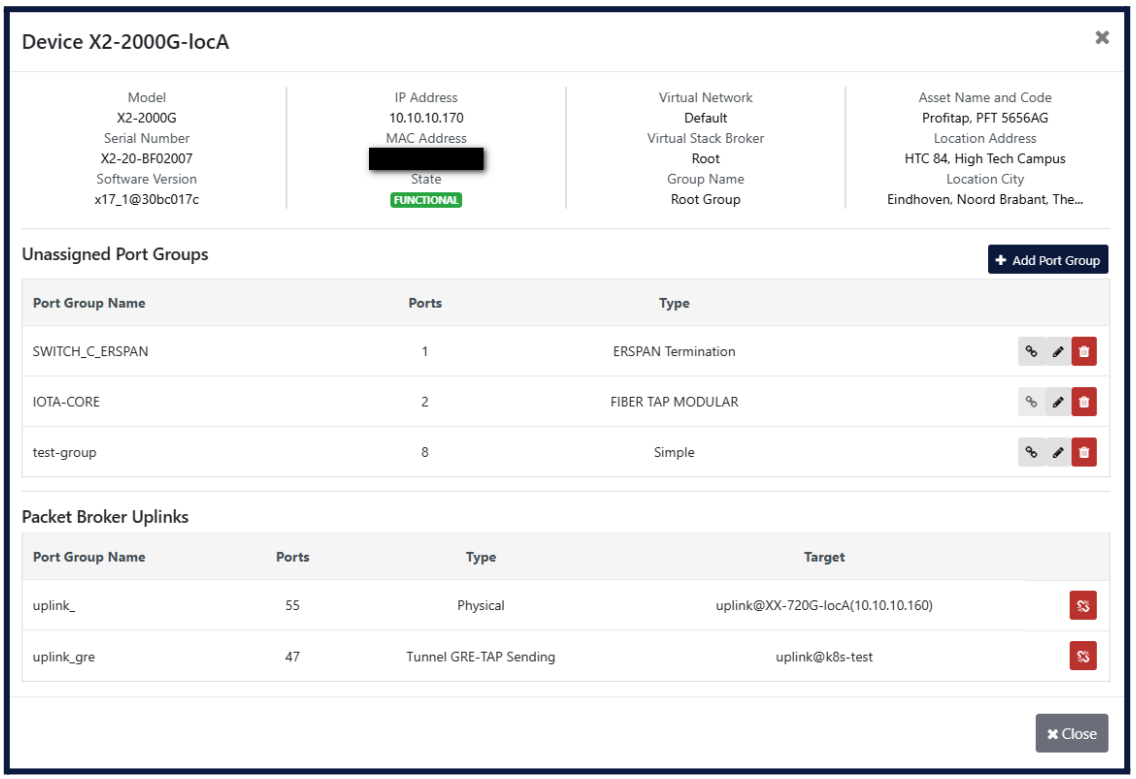
Add Group window

To create a network, click the *Add Network* button, and enter the network name in the *Add Virtual Network* window. The network's name can be changed at a later time by clicking the network's *Edit* button.

Add Virtual Network window

To remove a device, group or network, click its *Delete* button. If a group contains one or more devices, you will be asked whether these devices should be moved to another group, or removed along with the group.

Clicking on a device provides additional information about this device, and the ability to create port groups and packet broker uplinks prior to Traffic Management.

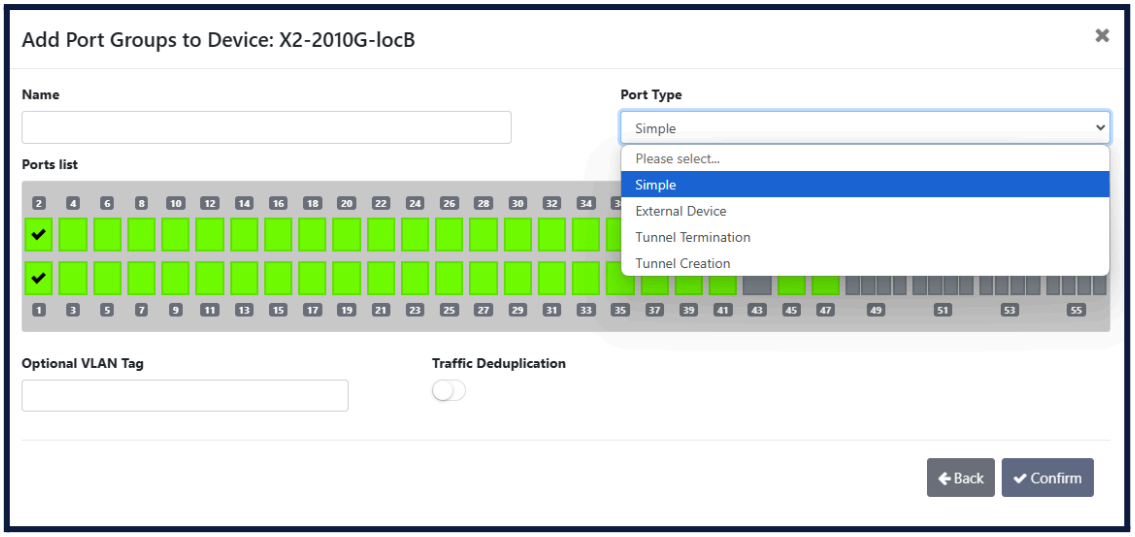


Device Details window

4.1.2. Port Groups

Profitap XX-Series and X2-Series packet brokers interfaces must be organized in port groups in order to create uplinks for the purpose of traffic management. A port group can contain one or more physical ports. Each physical port can only be used within a single port group. Port groups can be used to aggregate incoming traffic and/or distribute (load balance) the outgoing packets, deduplicate traffic, add a VLAN tag to the traffic, create an uplink to an external device, and create or terminate tunnels.

To create a port group, navigate to the **Device Manager > Registered Devices** page or **Traffic Management** page, click the device for which to create a port group to open its device details window, and click the *Add Port Group* button to open the *Add Port Groups* window. In this window, give a name to the port group, select the port type, select one or more ports, configure the additional options if necessary, and click the *Confirm* button.



Add Port Group window

The available port types are determined by the device type, and the additional options are determined by the device type and selected port type. Port types are as follows:

- **Simple:** Standard port group, used to create uplinks between devices managed by Supervisor. A VLAN tag can be added, and traffic deduplication can be enabled.
- **External Device:** Creates an uplink to a device that is not managed by Supervisor. The device type must be selected. This device type is strictly informational. A VLAN tag can be added, and traffic deduplication can be enabled.
- **Tunnel Termination:** Terminates an ERSPAN, GRE-TAP, or VXLAN tunnel. An IPv4 address and MAC address must be specified to associate to the port group. Can be used to receive traffic from K8s clusters and Azure VMs.
- **Tunnel Creation:** Encapsulates the traffic in an ERSPAN type 2, ERSPAN type 3, or GRE-TAP tunnel. The address fields and session ID or GRE key must be specified. The tunnel VLAN used in the tunnel can be specified. It is also possible to truncate the tunneled traffic by specifying the maximum size for each packet in bytes.

XX-Series devices can only create *Simple* and *External Device* port groups, without deduplication.
X2-Series devices can create all port group types, and use packet deduplication if available on the device.

Port groups that are not currently used in a packet broker uplink or that are connected to an external device are listed in the *Unassigned Port Groups* section of the device details window. From this listing, port groups can be edited or deleted, and they can be linked to a port group on another packet broker to create an uplink between the two.

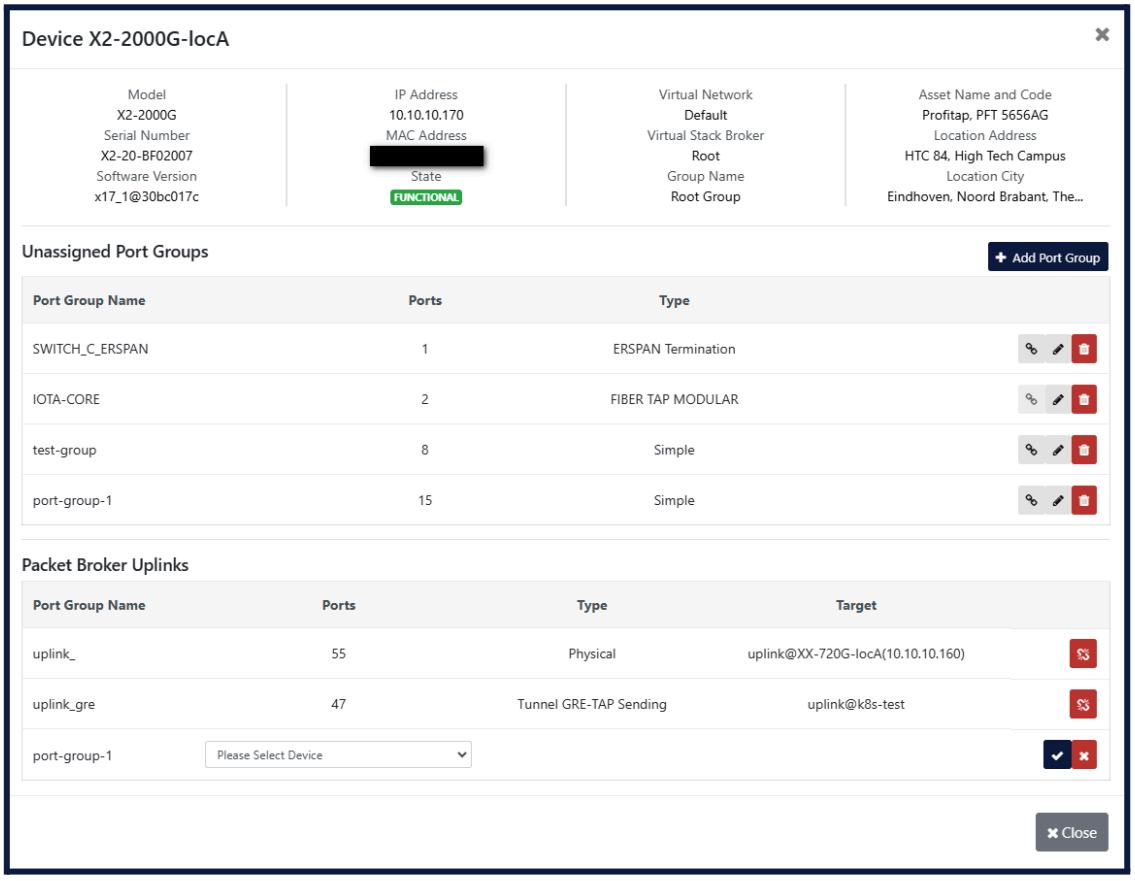
Note: Uplinks can only be created such that the devices or other entities in the virtual network are connected in some way. In other words, it is not possible to create separate clusters of devices with no connection between them within the same virtual network.

4.1.3. Packet Broker Uplink

Supervisor can help you monitor and control how the packet brokers hierarchy is interconnected. The physical connections between the packet brokers are called **uplinks**, and are used to distribute the traffic across the XX-Series or X2-Series fleet. Creating a simple uplink between two packet brokers is done by creating a port group of the *Simple* type on each device, then linking both port groups together. Port groups of the *Tunnel Termination* and *Tunnel Creation* types can also be used in packet broker uplinks.

The following process is an example for creating a simple uplink:

1. Navigate to the **Device Manager > Registered Devices** page.
2. Click one of the devices for which to create an uplink to open its device details window.
3. Click the **Add Port Group** button to open the *Add Port Groups* window.
4. Name the port group.
5. Set the *Port Type* to *Simple.*
6. Select the ports to include in the group.
7. (Optional) Enable traffic deduplication.
8. Click the *Confirm* button.
9. Repeat this process for the second device.
10. In the device details window of either device, click the *Link port group* button of the newly created port group to create a new entry in the *Packet Broker Uplinks* table below.
11. In this new entry, select the device to link and the port group on that device.
12. Click the *Add Uplink* button to confirm.



Device Details window

Add Port Groups window: creating a simple port group

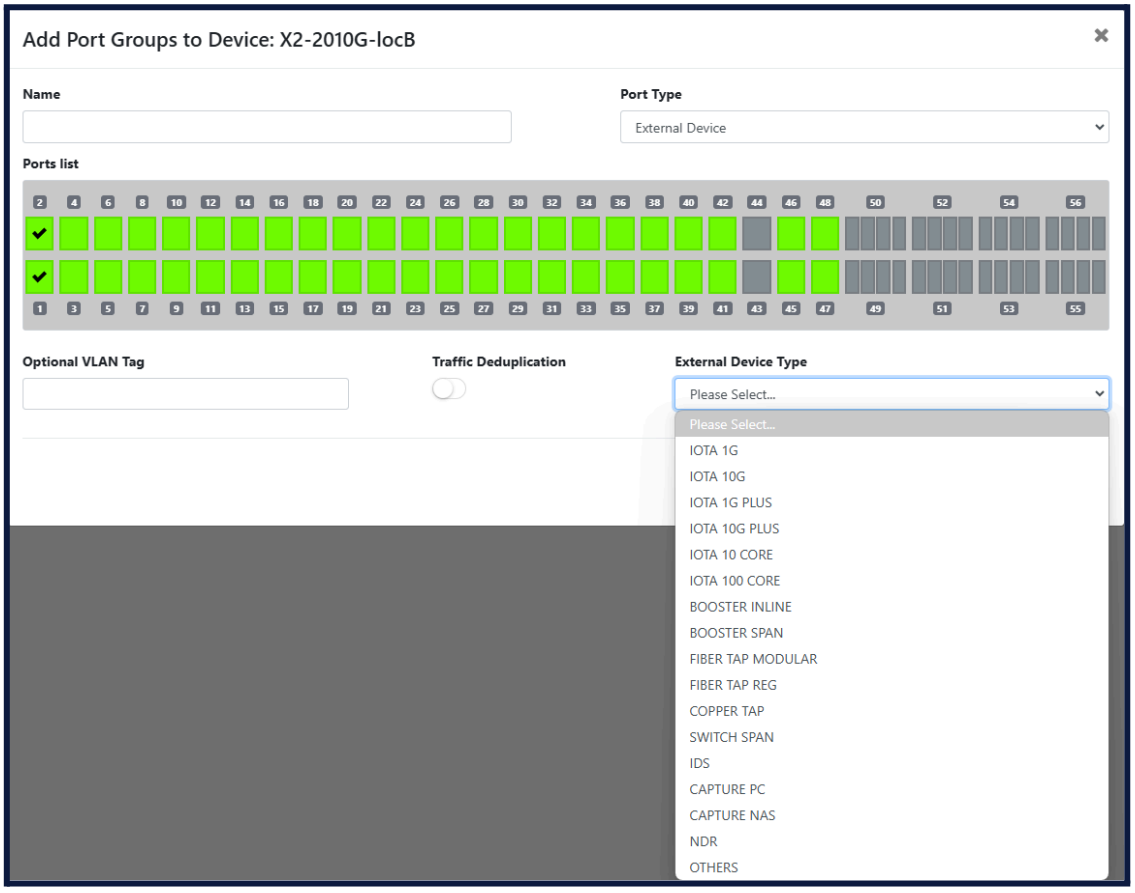
Note: The *Optional VLAN Tag* field has no effect if the port group is used in a packet broker uplink.

Once an uplink has been created, both devices will appear in the graphical view on the **Traffic Management** page.

4.1.4. External Device Uplink

Since Profitap packet brokers are likely not the only components of your visibility infrastructure, Supervisor allows you to map external devices connected to your XX-Series and X2-Series devices in the visibility network topology. These can be used as source or destination for your traffic rules. Adding an external device uplink is done by creating a port group of the *External Device* type.

1. Navigate to the **Device Manager > Registered Devices** page.
2. Click the device for which to create an external device uplink to open its device details window.
3. Click the **Add Port Group** button to open the *Add Port Groups* window.
4. Name the port group.
5. Set the *Port Type* to *External Device*.
6. Select the ports to include in the group.
7. Select the *External Device Type*.
8. (Optional) Add a VLAN tag. The value entered in this field will only take effect if the port group is used as a source in a traffic rule and the *Source VLAN Tag* option is enabled under *Advanced Options* in the rule creation menu (see Traffic Rules).
9. (Optional) Enable traffic deduplication.
10. Click the *Confirm* button.

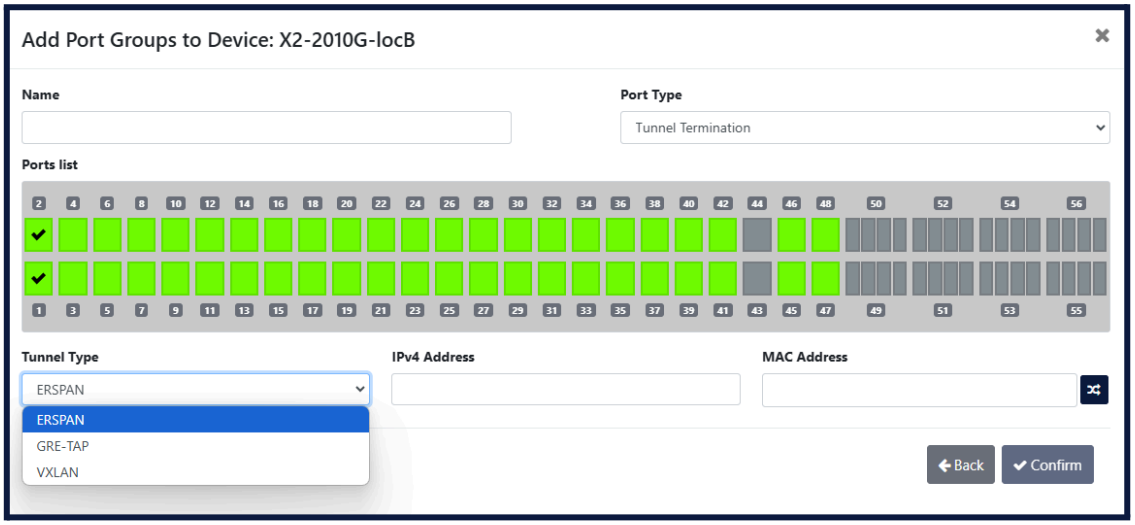


Once an external device uplink has been created, the external device will appear in the graphical view on the **Traffic Management** page.

4.1.5. Tunnel Termination

X2-Series devices can terminate ERSPAN, GRE-TAP, and VXLAN tunnels. This can be used for instance to receive traffic from K8s clusters and Azure VMs. This is done by creating a port group of the *Tunnel Termination* type.

1. Navigate to the **Device Manager > Registered Devices** page.
2. Click the device for which to create a *Tunnel Termination* port group to open its device details window.
3. Click the **Add Port Group** button to open the *Add Port Groups* window.
4. Name the port group.
5. Set the *Port Type* to *Tunnel Termination*.
6. Select the ports to include in the group.
7. Select the *Tunnel Type*.
8. Set an IPv4 and a MAC address for the interface (click the button next to the *MAC Address* field if you wish to generate a MAC address).
9. Click the *Confirm* button.

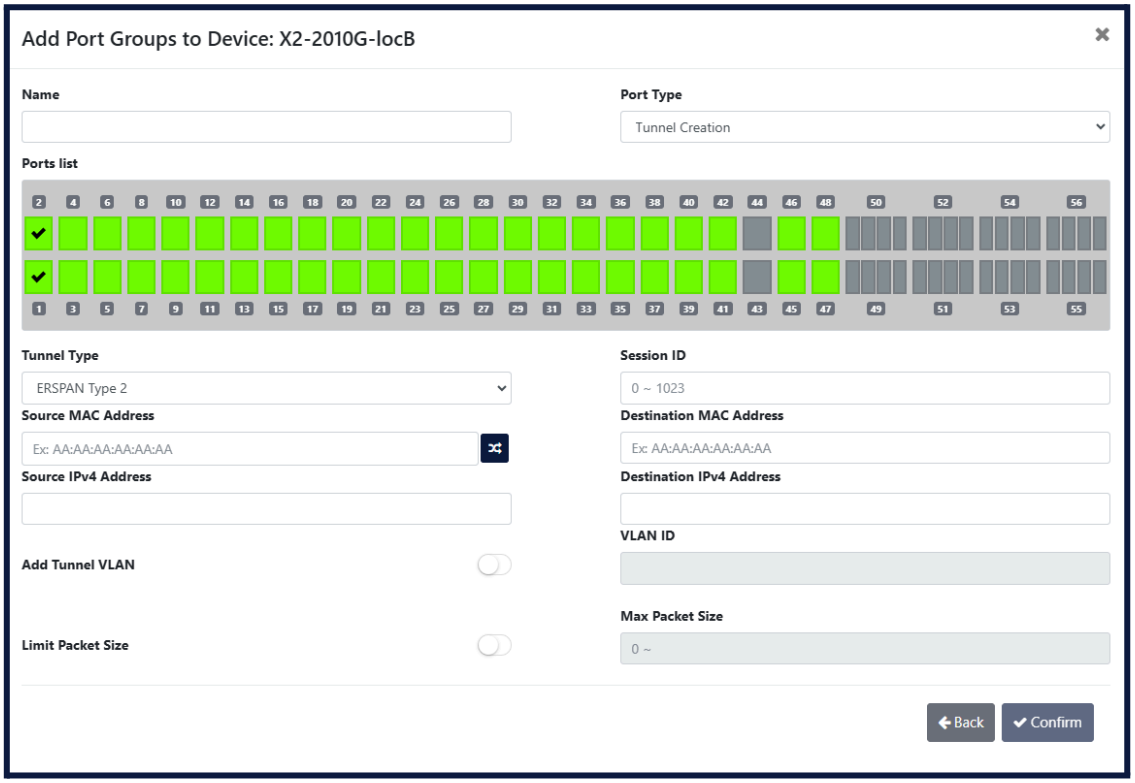


Once a tunnel termination port group has been created, the interface will appear in the graphical view on the **Traffic Management** page.

4.1.6. Tunnel Creation

X2-Series devices can encapsulate traffic in ERSPAN type 2, ERSPAN type 3, and GRE-TAP tunnels. This is done by creating a port group of the *Tunnel Creation* type.

1. Navigate to the **Device Manager > Registered Devices** page.
2. Click the device for which to create a *Tunnel Creation* port group to open its device details window.
3. Click the **Add Port Group** button to open the *Add Port Groups* window.
4. Name the port group.
5. Set the *Port Type* to *Tunnel Creation*.
6. Select the ports to include in the group.
7. Select the *Tunnel Type*.
8. Set the source and destination IPv4 and MAC addresses.
9. Set the session ID or GRE key.
10. (Optional) Enable *Add Tunnel VLAN* and specify the VLAN ID.
11. (Optional) Enable *Limit Packet Size* and specify the maximum packet size in bytes for truncating the tunneled traffic.
12. Click the *Confirm* button.



Once a tunnel creation port group has been created, the interface will appear in the graphical view on the **Traffic Management** page.

4.2. Traffic Statistics

The **Traffic Statistics** tab provides an overview of the traffic statistics of the devices managed by the Supervisor.

The search bar can be used to filter the current view to display specific devices or groups.

Clicking on a device adds it to, or removes it from, the statistics view in the bottom half of the page. Clicking on a group navigates to this group, listing the devices it contains, and allowing these devices to be added to, or removed from, the statistics view. Clicking the *Home* button navigates back to the root. Right-clicking a group allows a statistics column for this group to be added to, or removed from, the statistics view. Statistics columns can also be removed from the statistics view by clicking the *Clear statistics* button next to the column's name.

4.3. Firmware Update

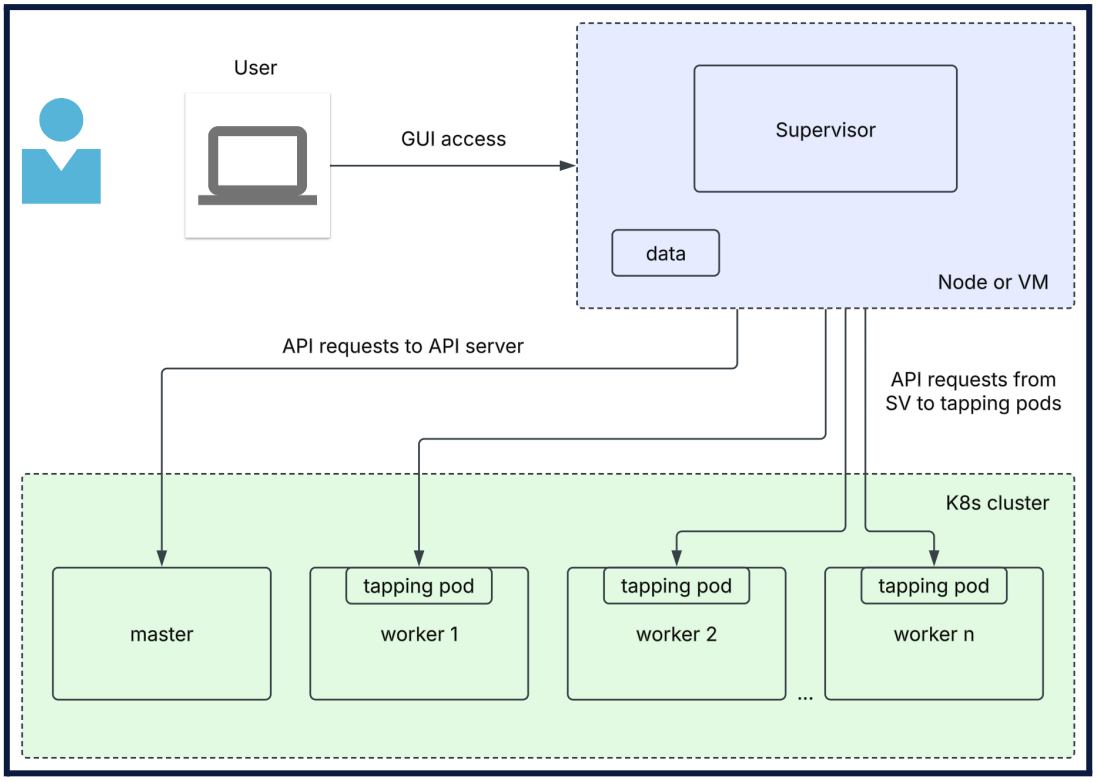
The **Firmware Update** tab allows firmware updates to be pushed to multiple devices at once.

Select the devices you would like to update from the list. The list can be filtered by device family, and by group. To filter by device family, use the *Device family* drop-down menu at the top left of the page. To filter by group, click the *filter by group* button next to the name of the group on the right-hand side of the list. To remove the group filter, click the *clear group filter* button at the top right of the list. To select or unselect all devices in the current view, click the checkbox at the top left of the list.

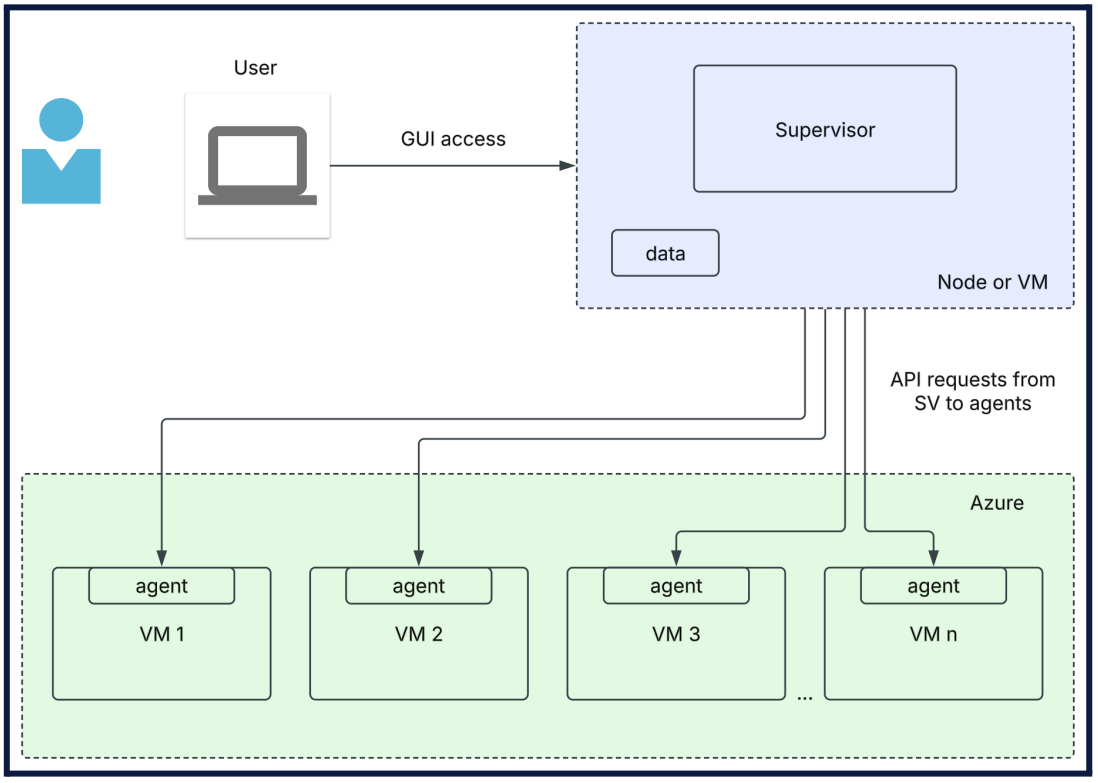
After having selected the appropriate devices, click the *Firmware update* button to select the firmware file. After confirming the update, the file will be uploaded to the Supervisor, after which the Supervisor will push the update to each of the selected devices. The update status can be followed on this page. Note that the current batch must be completed before a new batch can be started. Also note that, if attempting to update an XX-Series device using an X2-Series firmware file (or vice versa), the update will fail for that particular device.

5. Cloud TAP

This chapter describes the Supervisor section specific to the Cloud TAP module, accessed via the *Cloud TAP* menu item, as pertains to tapping Kubernetes clusters and Azure Virtual Machines (VMs). Kubernetes clusters and Azure VMs can be managed further in the sections described in the Traffic Management and Event Monitoring chapters.



Communication between Supervisor and K8s clusters



Communication between Supervisor and Azure VM environments

5.1. Registered Clusters

5.1.1. Overview

The **Registered Clusters** tab of the **Cloud TAP** page provides an overview of the Kubernetes and Azure VM environments managed by the Supervisor, general information about them, and their status.

Each virtual environment can be assigned to a Network. Networks can be used to isolate devices and virtual environments in logical networks in the Traffic Management page, with each Network operating with its own set of traffic rules.

Clicking on an environment provides additional information about this environment, and the ability to create pod groups or interface groups and tunnel destinations prior to Traffic Management.

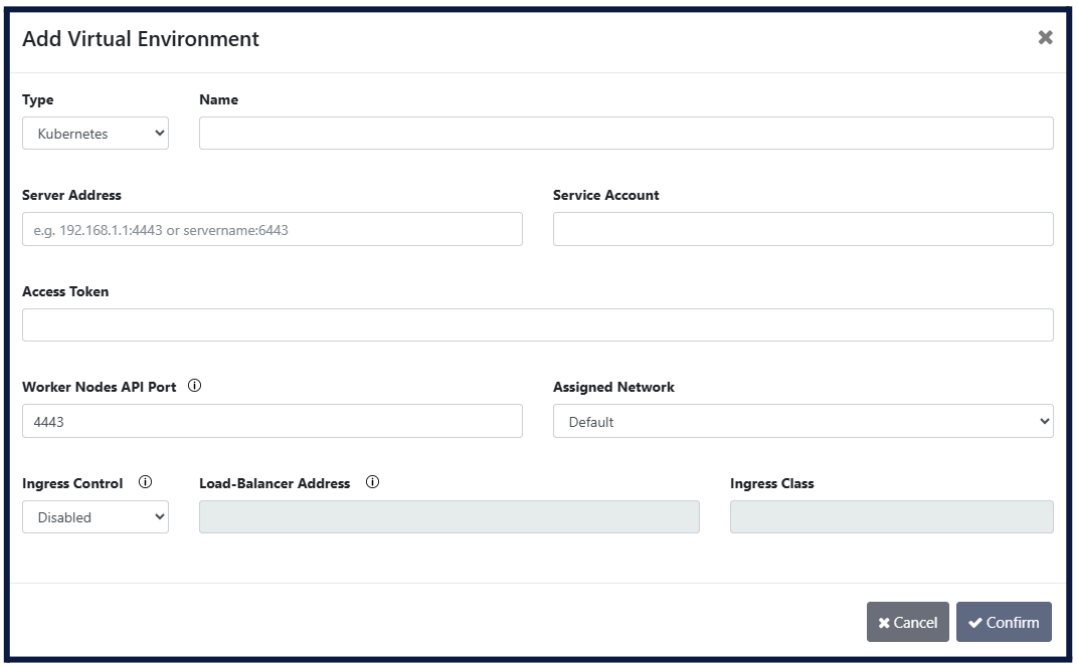
Clicking on a network navigates to this network, listing the environments it contains. Clicking the *Home* button navigates back to the root.

From this dashboard, environments and networks can be added, modified, or removed.

To add a new environment, click the *Add Virtual Environment* button in the top right corner of the interface, select Azure or Kubernetes and enter the environment's information in the *Add Virtual Environment* window. Select a network in this window to add the environment to this network. The environment's information can be changed at a later time by clicking the environment's *Edit* button.

To create a network, click the *Add Network* button, and enter the network name in the *Add Virtual Network* window. The network's name can be changed at a later time by clicking the network's *Edit* button.

5.1.2. Kubernetes - Adding a Virtual Environment



Add Virtual Environment window - Kubernetes

- **Type:** The type of environment [Azure/Kubernetes].
- **Name:** A name for the virtual environment.
- **Server Address:** The IP address or server name of the kubernetes API server. The L4 port can be specified if required by the server configuration.
- **Service Account:** The service account under which the token was created. This service account must be within the '`profitap`' namespace.
- **Access Token:** This token will be used to access the kubernetes master node. For creating the token, see Creating Token for Supervisor Access.
- **Worker Nodes API Port:** The port number on the worker node(s) that the tapping pod will listen on. Supervisor will communicate with the tapping pod through this port, so it should be made accessible and not blocked.
- **Assigned Network:** The virtual network that the virtual environment will be assigned to.
- **Ingress Control:** In case an ingress controller is deployed on the kubernetes cluster.
    - **Disabled:** No ingress controller.
    - **Default:** Default ingress class configured on the cluster.
    - **Custom:** Specific ingress class associated with the chosen ingress controller.
- **Load-Balancer Address:** The IP address or server name of the load balancer service. The L4 port can be specified if necessary.
- **Ingress Class:** Name of the ingress class (e.g. nginx).

The default ingress class can be viewed using the following command (if configured):

```
kubectl get ingressclass -o=jsonpath='{range
.items[*]}{.metadata.name}{"\t"}{.metadata.annotations.ingressclass\.kubernetes\.
io/is-default-class}{"\n"}{end}'
```

5.1.3. Kubernetes - Creating Token for Supervisor Access

Create the following YAML file (e.g. required-rights.yml):

```yaml
apiVersion: v1
kind: Namespace
metadata:
  name: profitap

---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: my-sa1
  namespace: profitap

---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: my-cluster-role
rules:
- apiGroups: [""]
  resources: ["pods"]
  verbs: ["delete", "list", "create"]
- apiGroups: [""]
  resources: ["nodes"]
  verbs: ["list"]
- apiGroups: [""]
  resources: ["services", "namespaces"]
  verbs: ["delete", "create"]
```

```
  - apiGroups: [""]
    resources: ["secrets"]
    verbs: ["get", "delete", "create"]
  - apiGroups: [""]
    resources: ["serviceaccounts/token"]
    verbs: ["create"]
  - apiGroups: ["networking.k8s.io"]
    resources: ["ingresses"]
    verbs: ["create", "delete"]
  - apiGroups: ["networking.k8s.io"]
    resources: ["ingressclasses"]
    verbs: ["list"]

  ---
  apiVersion: rbac.authorization.k8s.io/v1
  kind: ClusterRoleBinding
  metadata:
    name: my-cluster-role-binding
  subjects:
  - kind: ServiceAccount
    name: my-sa1
    namespace: profitap
  roleRef:
    kind: ClusterRole
    name: my-cluster-role
    apiGroup: rbac.authorization.k8s.io

  ---
  apiVersion: v1
  kind: Secret
  metadata:
    name: my-sa1-token
    annotations:
      kubernetes.io/service-account.name: my-sa1
    namespace: profitap
  type: kubernetes.io/service-account-token
```

Run the file. It will create a service account with the required rights, secret and token.

```
kubectl apply -f required-rights.yml
```

Extract the bearer token from a Kubernetes Secret that stores a ServiceAccount token.

```
kubectl get secret my-sa1-token -n profitap -o jsonpath='{.data.token}' | base64
--decode
```

The following command is an alternative to the one above for extracting the bearer token:

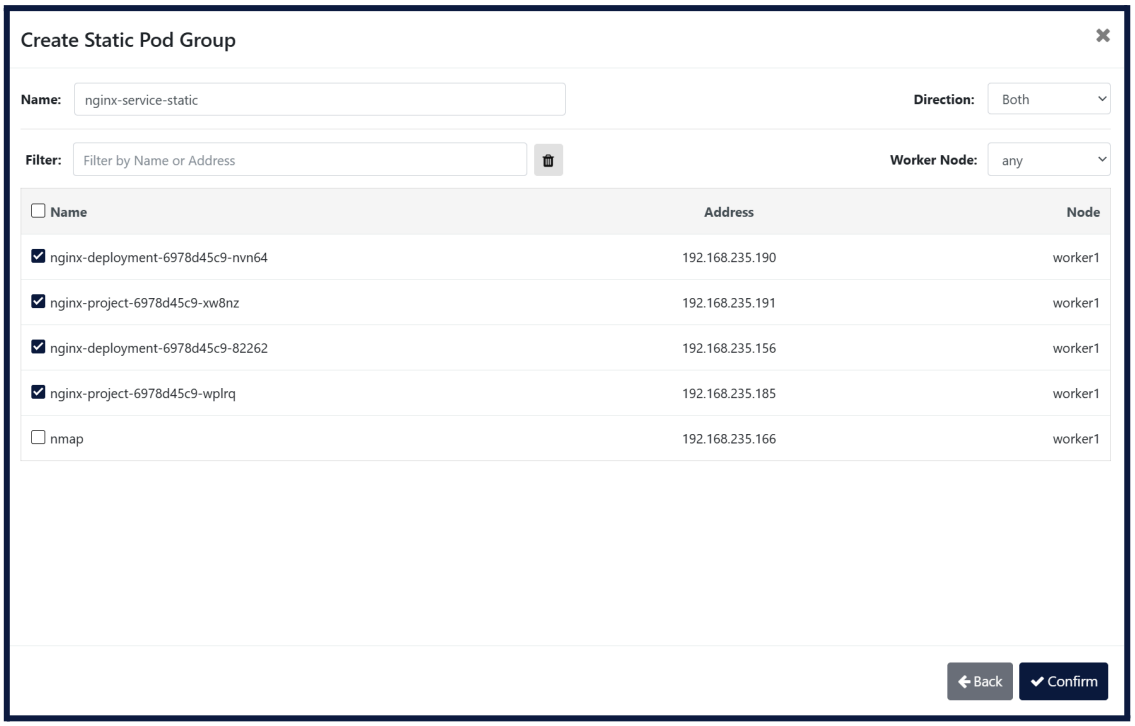
```
kubectl describe secret my-sa1-token -n profitap
```

5.1.4. Kubernetes - Source Pod Groups

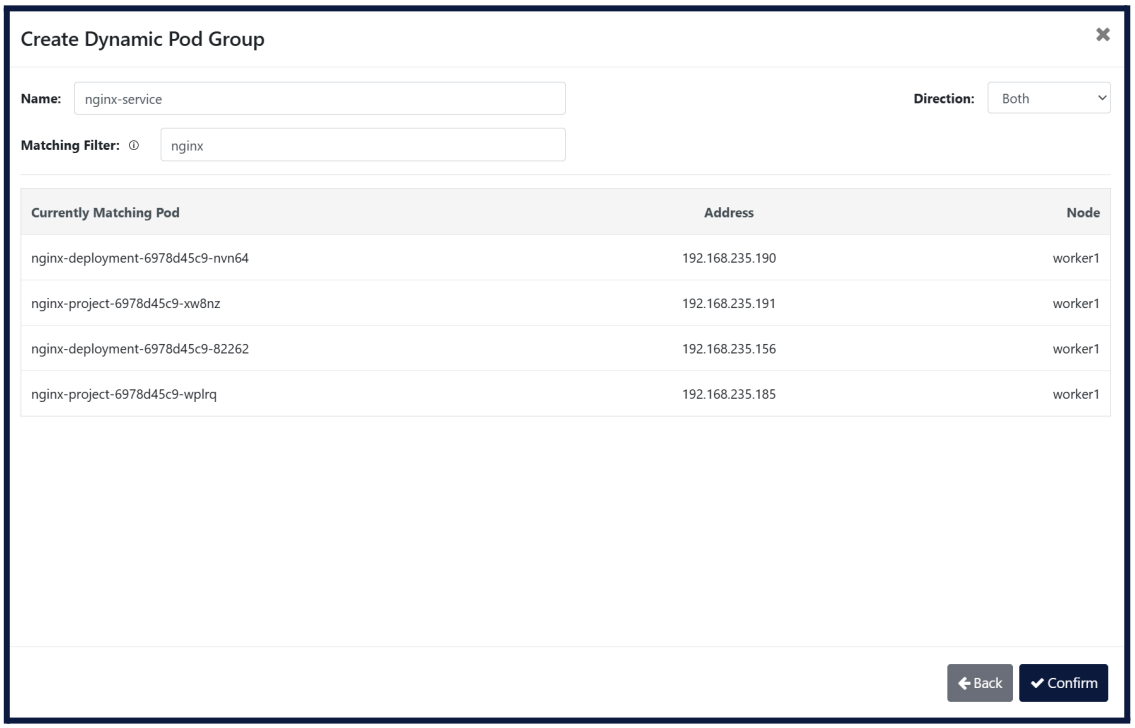
Pod Groups are groups of Kubernetes pods monitored by Cloud TAP and used as a source of traffic to be sent to analysis tools. Two types of pod groups can be created: Static and Dynamic. **Static Pod Groups** contain specific pods that were manually selected. **Dynamic Pod Groups** contain any pods matching a name filter, and are automatically updated to include new pods which name matches that filter.

To create a pod group on a Kubernetes cluster, navigate to the **Cloud TAP > Registered Clusters** page, click a Kubernetes cluster to open its details window, and click the *Create Static Pod Group* or *Create Dynamic Pod Group* button:

- For **Static Pod Groups**, set a name, select which traffic direction to monitor (ingress, egress, or both), and select the pods to include in the group.
- For **Dynamic Pod Groups**, set a name, select which traffic direction to monitor (ingress, egress, or both), and set a *Match Filter* to match the name of the pods to automatically include in the group.



Create Static Pod Group window



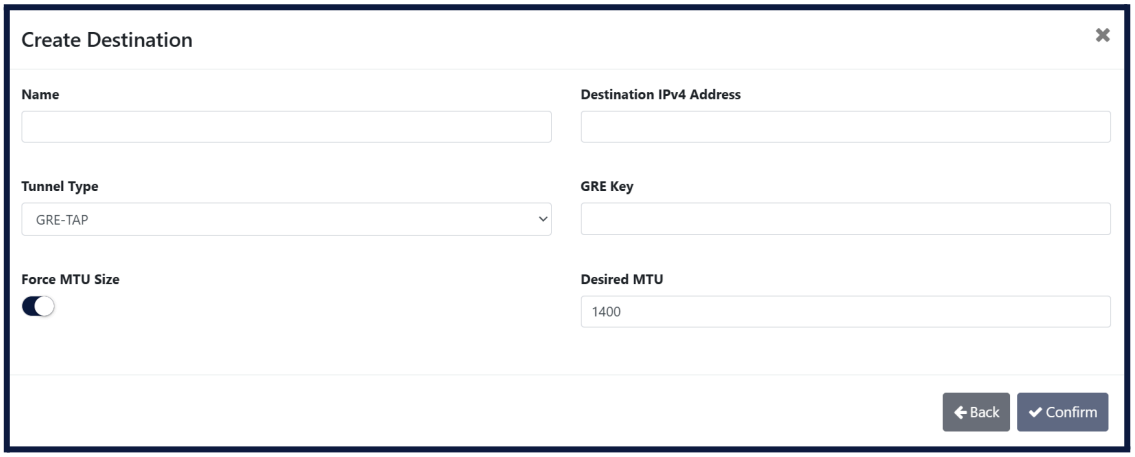
Create Dynamic Pod Group window

5.1.5. Kubernetes - Tunnel Destinations

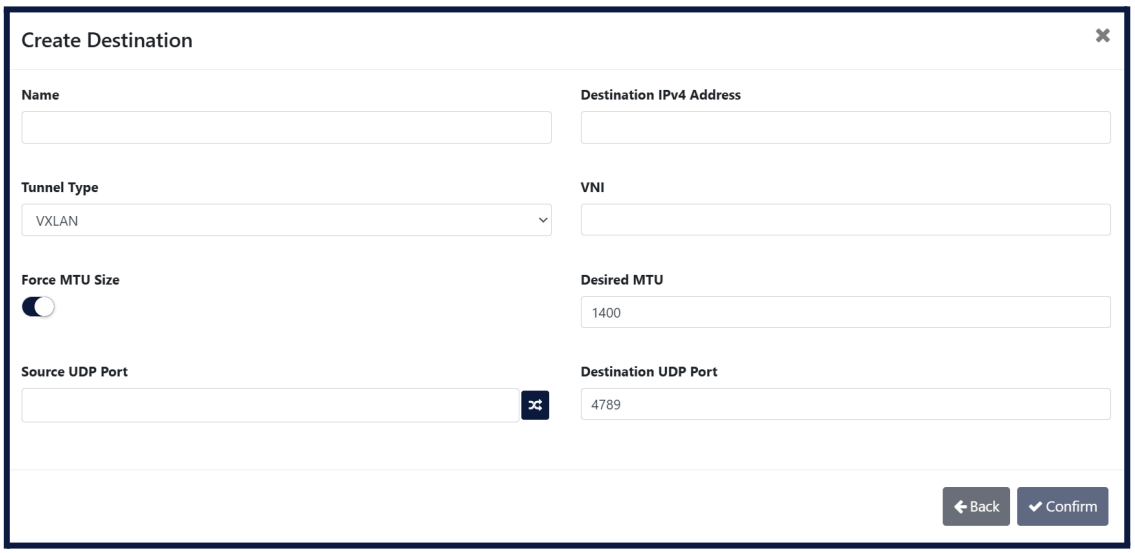
Tunnel Destinations define where the monitored traffic from the configured pod groups will be sent, encapsulated in a GRE-TAP, VXLAN, or ERSPAN tunnel. This can for instance be an X2-Series device on which a Tunnel Termination port group was created, from where it can then be forwarded to analysis tools via Traffic Rules.

Tunnel destination creation process:

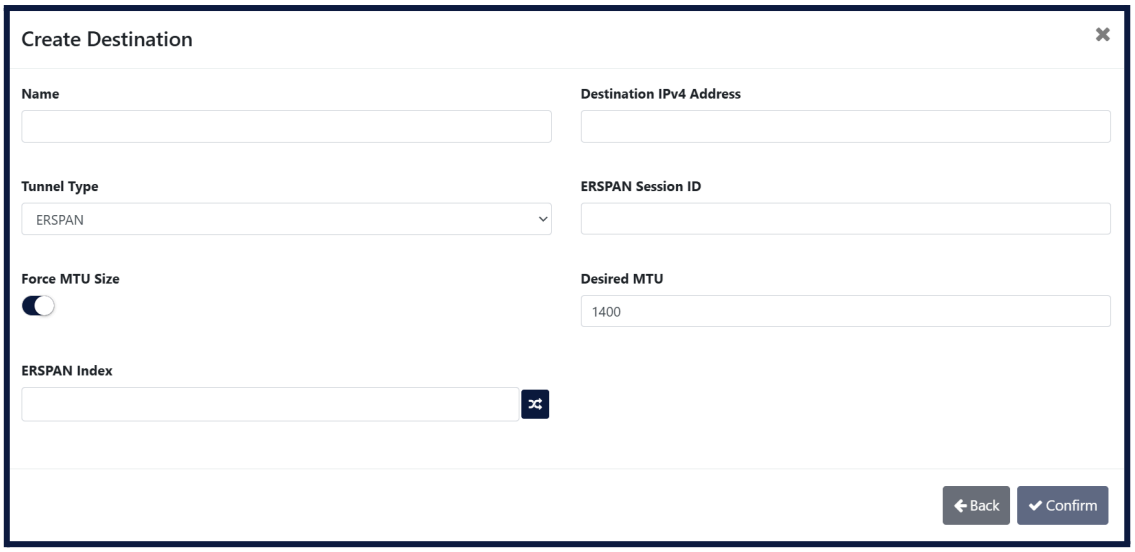
1. Navigate to the **Cloud TAP > Registered Clusters** page.
2. Click a Kubernetes cluster to open its details window.
3. Click the *Create Destination* button.
4. Set a name.
5. Set a destination IPv4 address.
6. Select the tunnel type (GRE-TAP/VXLAN/ERSPAN).
7. (For GRE-TAP) Set a GRE key.
8. (For VXLAN) Set a VNI, source UDP port, and destination UDP port.
9. (For ERSPAN) Set an ERSPAN session ID and ERSPAN index.
10. (Optional) Enable *Force MTU Size* and set the desired MTU.
11. Click the *Confirm* button.



Creating a GRE-TAP tunnel destination on a K8s cluster



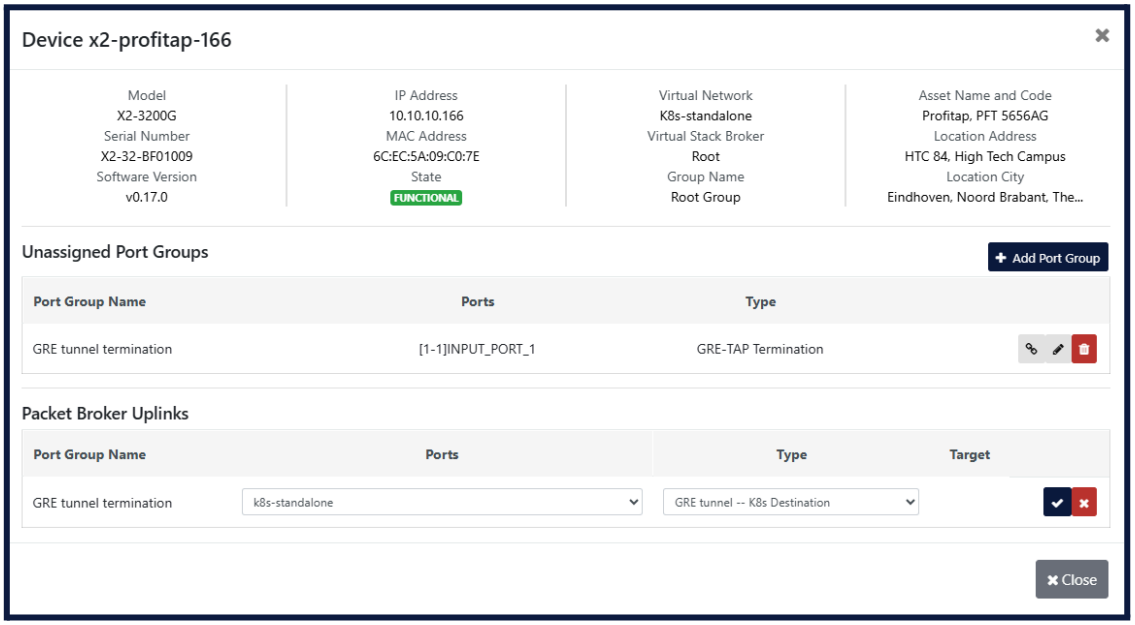
Creating a VXLAN tunnel destination on a K8s cluster



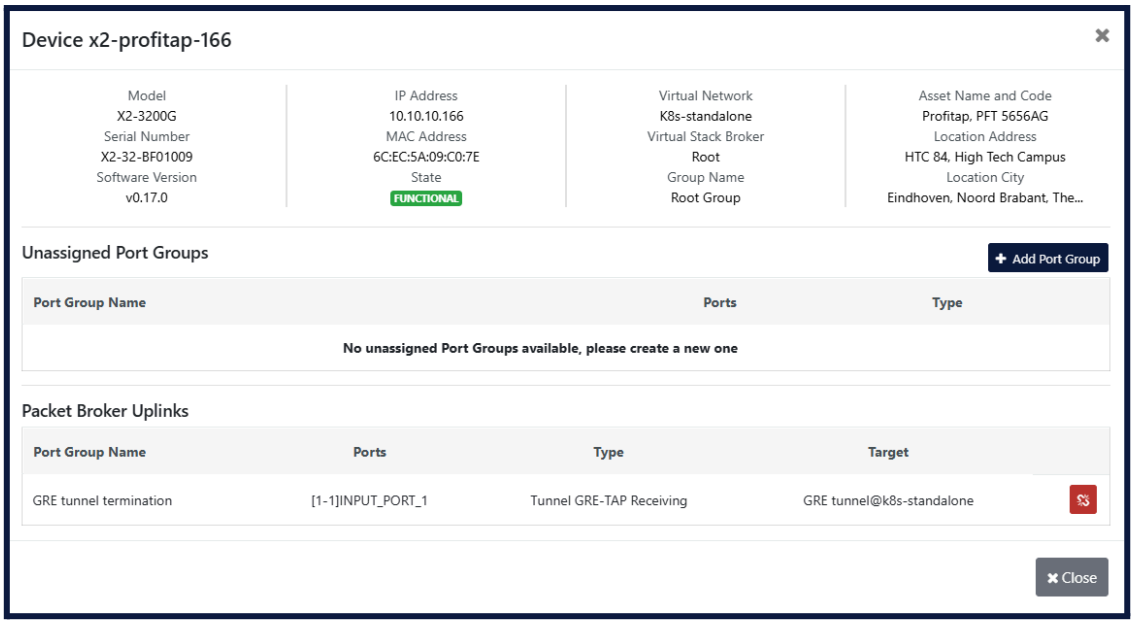
Creating an ERSPAN tunnel destination on a K8s cluster

You can then create Traffic Rules to send traffic from specific pod groups to the tunnel destination configured above.

To link a K8s cluster to an X2-Series device, create a Tunnel Termination port group on that device with the same IPv4 address as the one specified above, then create an uplink between this port group and the tunnel destination configured above. You can then create Traffic Rules to send traffic from specific pod groups to any destination linked through the X2-Series device.

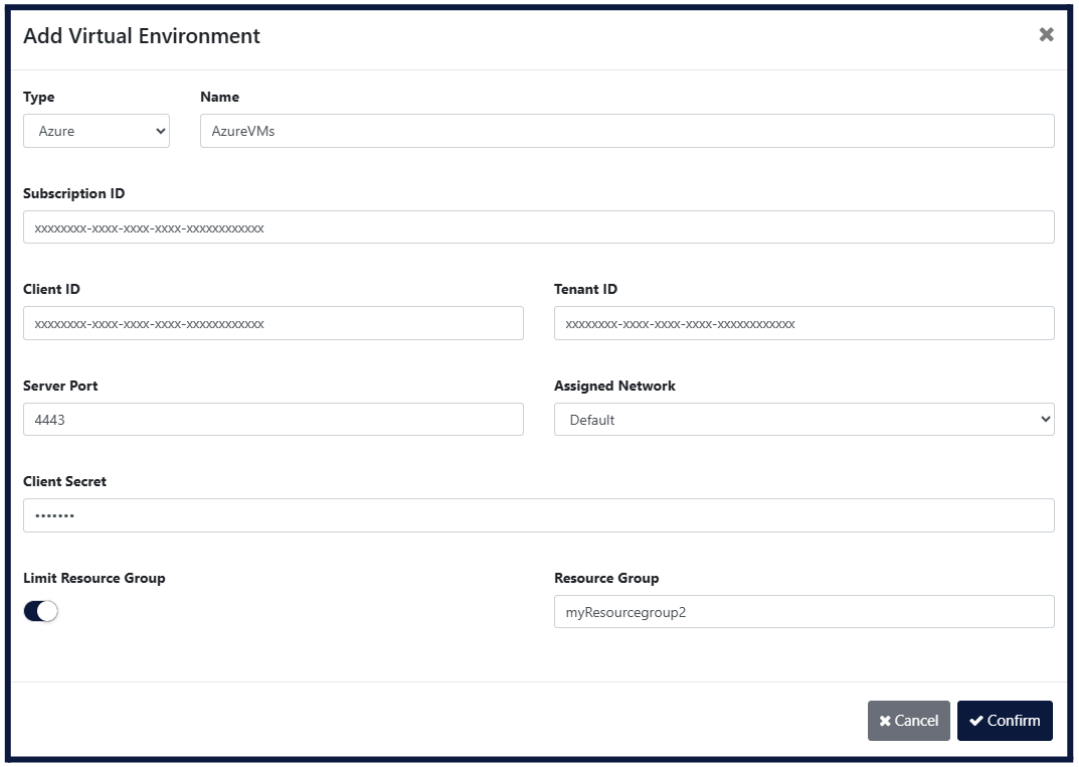


Creating an uplink between a K8s tunnel destination and a Tunnel Termination port group on an X2-Series device



Confirming the uplink creation displays the uplink in the list of packet broker uplinks and removes the port group used from the list of unassigned port groups

5.1.6. Azure - Adding a Virtual Environment



Add Virtual Environment window - Azure

- **Type:** The type of environment [Azure/Kubernetes].
- **Name:** A name for the virtual environment.
- **Subscription ID:** Unique ID of your Azure subscription.
- **Client ID:** Application ID of a service principal/app used to access Azure. For creating the Application ID, see Creating Application ID and assigning required roles.
- **Tenant ID:** ID of the Microsoft Entra ID tenant/organization that owns the subscription.

- **Server Port:** TCP/UDP port the VM listens on (through the agent). Supervisor will use this port to communicate with the agent, so it should be made accessible and not blocked.
- **Assigned Network:** The virtual network that the virtual environment will be assigned to.
- **Client Secret:** Secret password for the service principal/app.
- **Limit Resource Group:** Enable to only display resources in the specified resource group.
- **Resource Group:** Logical collection that groups related VM resources. If *Limit Resource Group* is enabled, specify the resource group here.

5.1.7. Azure - Creating Application ID and assigning required roles

The Application ID is referred to as Client ID in the Supervisor UI. This chapter uses the terms interchangeably. The Application ID must have the following roles:

- **Contributor Role**
  This role enables the Supervisor (SV) to create virtual environments on Azure via the Application ID (Client ID). The scope of this role can be either a Subscription or a Resource Group.

- **Storage Blob Data Contributor**
  This role allows the SV to create storage accounts for traffic mirroring for Windows VMs via the Application ID. If no Windows VMs are targeted by Supervisor, this role may be omitted. The scope can be a Subscription, a Resource Group, or a Resource Group prefixed with '`profitap`'.

- **Key Vault Secrets Officer**
  This role enables the SV to create Key Vaults for Linux VMs through the Application ID. If no Linux VMs are targeted by Supervisor, this role may be omitted. The scope can be a Subscription, a Resource Group, or a Resource Group prefixed with '`profitap`'.

- **Key Vault Data Access Administrator**
  This role allows the SV to assign roles on the Key Vault for Linux VMs via the Application ID. If no Linux VMs are targeted by Supervisor, this role may be omitted. The scope can be a Subscription, a Resource Group, or a Resource Group prefixed with '`profitap`'.

Create a new App Registration (Application ID or Client ID)

```
az ad app create --display-name <app name> --query appId -o tsv
```

Sample command:

```
az ad app create --display-name myapp1 --query appId -o tsv
```

The command above creates an Application ID named '`myapp1`' and displays its ID. Copy this ID. For our purposes, we will assume the ID is '`c0303030-0303-0303-0303-030303030303`'. This ID will be used as an example for simplicity.

Use the following command to see the application details:

```
az ad app show \
--id <app id> \
--query "{displayName:displayName, appId:appId, objectId:id}" -o table
```

Sample command:

```
az ad app show \
--id c0303030-0303-0303-0303-030303030303 \
--query "{displayName:displayName, appId:appId, objectId:id}" -o table
```

Create the Service Principal for the application

```
az ad sp create --id <app id>
```

Sample command:

```
az ad sp create --id c0303030-0303-0303-0303-030303030303
```

Create a client secret (with 1-year expiry date)

```
az ad app credential reset --id <app id> --display-name myapp1-secret --years 1
```

Sample command:

```
az ad app credential reset \
--id c0303030-0303-0303-0303-030303030303 \
--display-name myapp1-secret \
--years 1
```

The command above will display the password. Make sure to save it, as it cannot be viewed later. We will use this password in Supervisor during the creation of the virtual environment.

Assign the Contributor role to the Application ID

The scope should be either the Subscription or a Resource Group. For this example, we will assume the Subscription ID is '`B0202020-0202-0202-0202-020202020202`'. Without this role, an Azure virtual environment cannot be created on Supervisor.

```
az role assignment create \
--assignee <app id> \
--role Contributor \
--scope /subscriptions/<subscription id>
```

Sample command (scope is Subscription):

```
az role assignment create \
--assignee c0303030-0303-0303-0303-030303030303 \
--role Contributor \
--scope /subscriptions/b0202020-0202-0202-0202-020202020202
```

Sample command (scope is a Resource Group):

```
az role assignment create \
--assignee c0303030-0303-0303-0303-030303030303 \
--role Contributor \
--scope
/subscriptions/b0202020-0202-0202-0202-020202020202/resourceGroups/myresourcegrou
p1
```

Verify the role assignment:

```
az role assignment list \
--assignee c0303030-0303-0303-0303-030303030303 \
--all -o table
```

Create a Resource Group (RG) for holding the Key Vault and/or storage accounts

The Resource Group's name should start with '`profitap`'. The naming is not case-sensitive.

```
az group create --name <profitap resource group> --location <location>
```

Sample command:

```
az group create --name profitap-rg1 --location northeurope
```

Verify the Resource Group:

```
az group show --name profitap-myrg1 -o table
```

Supervisor uses Key Vault for Linux VMs, and storage account for Windows VMs.

Creating a Resource Group (RG) whose name begins with '`profitap`' (case-insensitive) is not mandatory, but strongly recommended. Using a dedicated RG allows Supervisor to keep its components (Key Vault and storage accounts) organized in one place. During tapping-agent deployment to VMs, Supervisor searches for an RG whose name starts with '`profitap`' and will create the Key Vault and storage accounts inside of that RG.

If a dedicated “profitap” RG is not present, Supervisor will create the Key Vault and storage accounts in the alphabetically first existing RG.

Assign roles to the Service Principal at the Resource Group, Subscription, or profitap-RG scope

Sample commands (the scope is profitap-RG):

```
az role assignment create \
--assignee c0303030-0303-0303-0303-030303030303 \
--role "Key Vault Secrets Officer" \
--scope
/subscriptions/b0202020-0202-0202-0202-020202020202/resourceGroups/profitap-rg1
```

```
az role assignment create \
--assignee c0303030-0303-0303-0303-030303030303 \
--role "Key Vault Data Access Administrator" \
--scope
/subscriptions/b0202020-0202-0202-0202-020202020202/resourceGroups/profitap-rg1
```

```
az role assignment create \
--assignee c0303030-0303-0303-0303-030303030303 \
--role "Storage Blob Data Contributor" \
--scope
/subscriptions/b0202020-0202-0202-0202-020202020202/resourceGroups/profitap-rg1
```

Alternatively, the scope in the above role assignments could be the whole Subscription or a specific Resource Group.

```
--scope /subscriptions/b0202020-0202-0202-0202-020202020202
```

```
--scope /subscriptions/e0b37afa-64a0-4036-89e1-5bdc2dd02f14/resourceGroups/rg2
```

Verify that the application has the required roles. Four roles must be assigned to the application for Windows and Linux VMs.

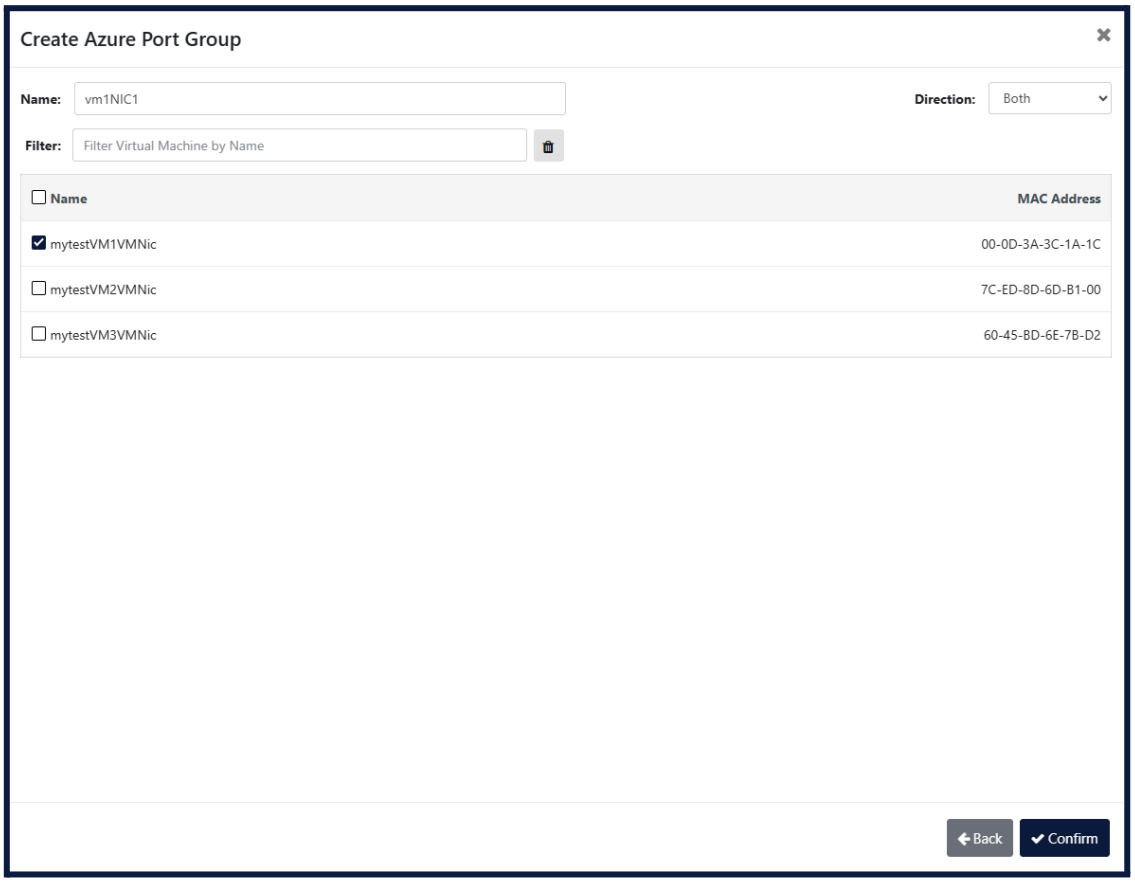
```
az role assignment list \
--assignee c0303030-0303-0303-0303-030303030303 \
--all -o table
```

5.1.8. Azure - Interface Groups

Interface Groups are groups of Azure VM network interfaces (NICs) monitored by Cloud TAP and used as a source of traffic to be sent to analysis tools.

Interface group creation process:

1. Navigate to the **Cloud TAP > Registered Clusters** page.
2. Click an Azure VM environment to open its details window.
3. Click the *Create Interface Group* button.
4. Set a name.
5. Select which traffic direction to monitor (ingress, egress, or both).
6. Select the VM NICs to include in the group.
7. Click the *Confirm* button.



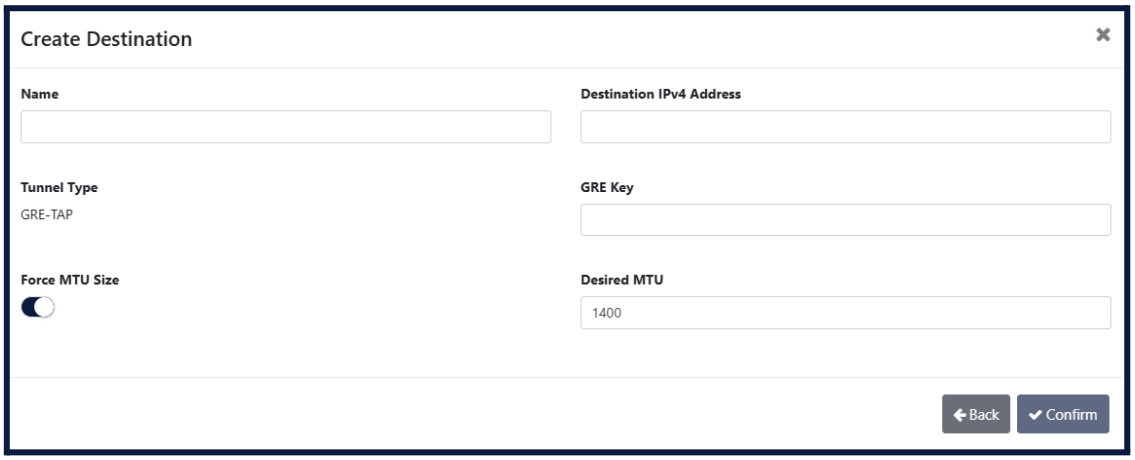
Create Azure Port Group window

5.1.9. Azure - Tunnel Destinations

Tunnel Destinations define where the monitored traffic from the configured interface groups will be sent, encapsulated in a GRE-TAP tunnel. This can for instance be an X2-Series device on which a Tunnel Termination port group was created, from where it can then be forwarded to analysis tools via Traffic Rules.

Tunnel destination creation process:

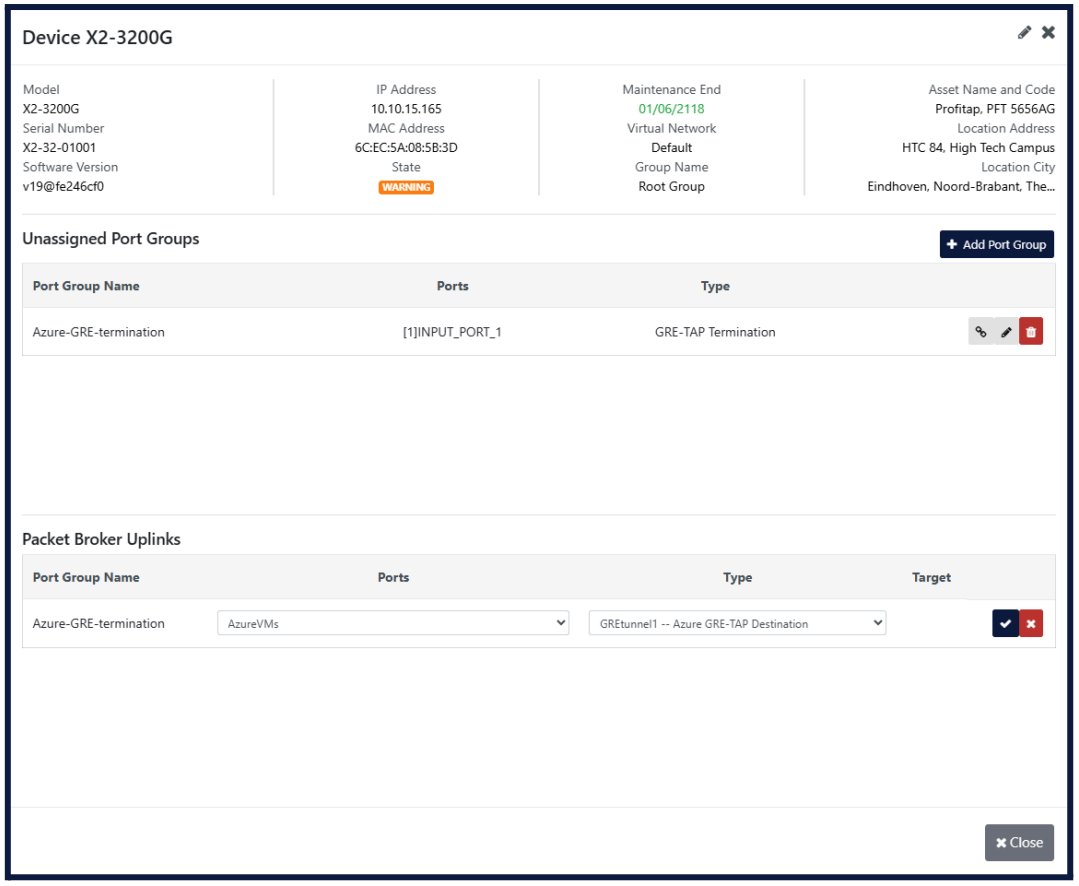
1. Navigate to the **Cloud TAP > Registered Clusters** page.
2. Click an Azure VM environment to open its details window.
3. Click the *Create Destination* button.
4. Set a name.
5. Set a destination IPv4 address.
6. Set a GRE key.
7. (Optional) Enable *Force MTU Size* and set the desired MTU.
8. Click the *Confirm* button.



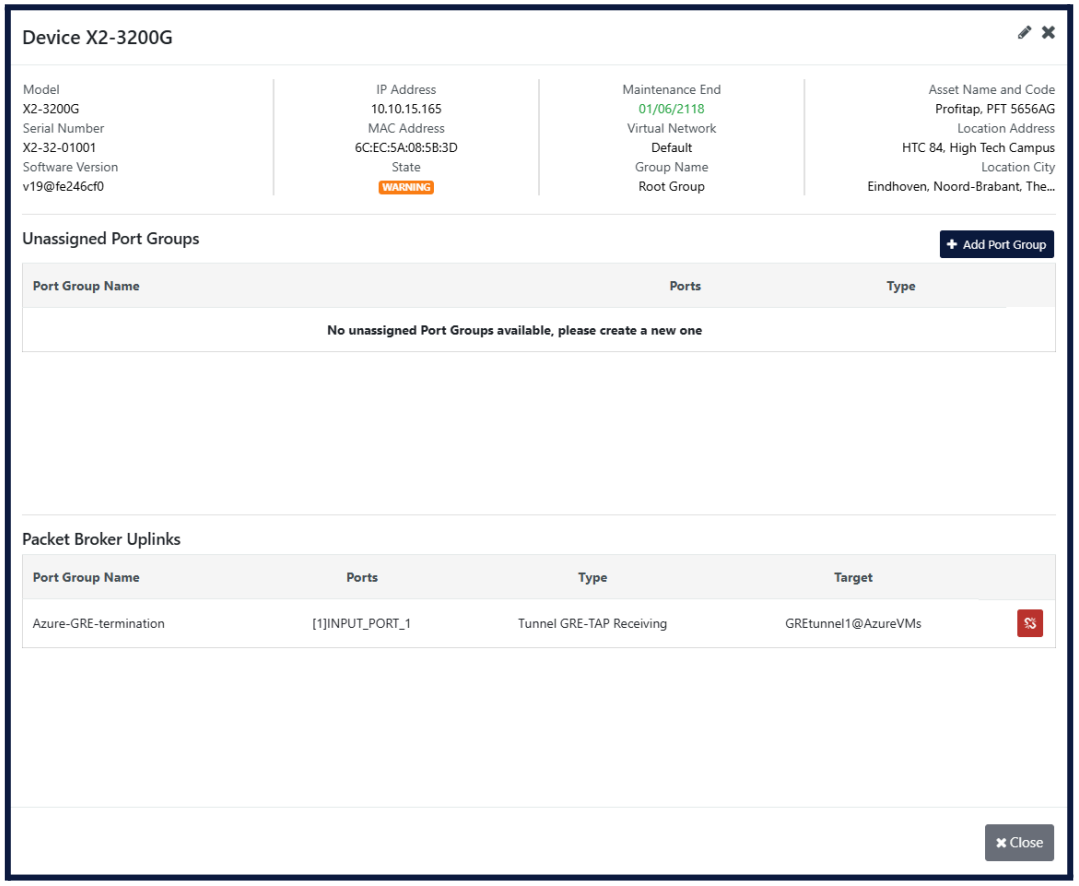
Creating a GRE-TAP tunnel destination on an Azure VM environment

You can then create Traffic Rules to send traffic from specific interface groups to the tunnel destination configured above.

To link an Azure VM environment to an X2-Series device, create a Tunnel Termination port group on that device with the same IPv4 address as the one specified above, then create an uplink between this port group and the tunnel destination configured above. You can then create Traffic Rules to send traffic from specific interface groups to any destination linked through the X2-Series device.



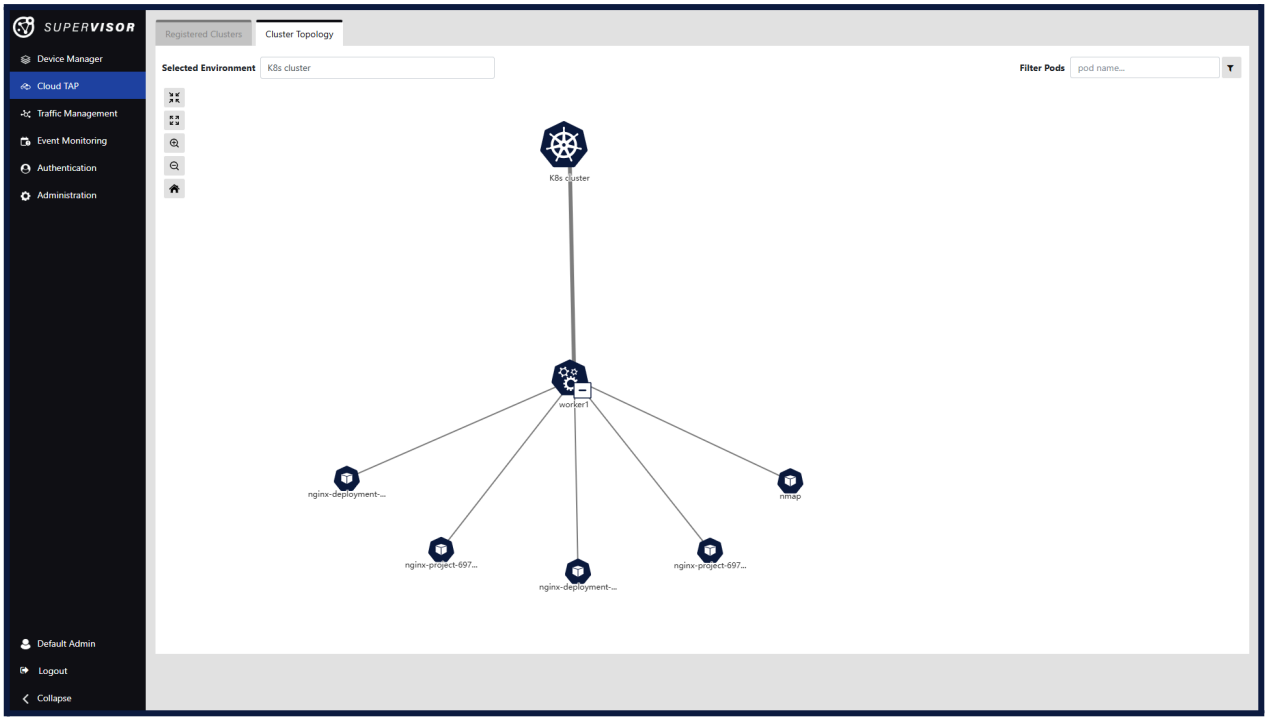
Creating an uplink between an Azure tunnel destination and a Tunnel Termination port group on an X2-Series device



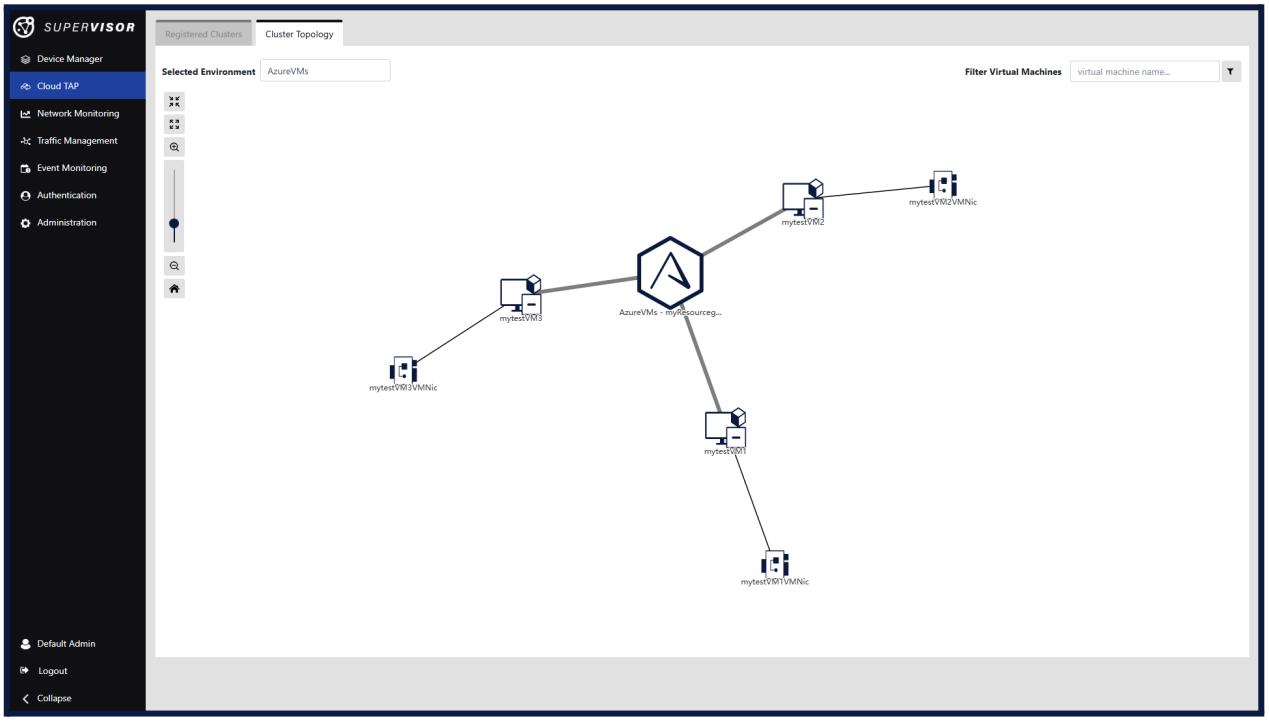
Confirming the uplink creation displays the uplink in the list of packet broker uplinks and removes the port group used from the list of unassigned port groups

5.2. Cluster Topology

The **Cluster Topology** tab of the **Cloud TAP** page gives a view of the topology of registered Kubernetes clusters and Azure VM environments. Select the environment to view by selecting it in the *Selected Environment* drop-down menu in the top left corner.



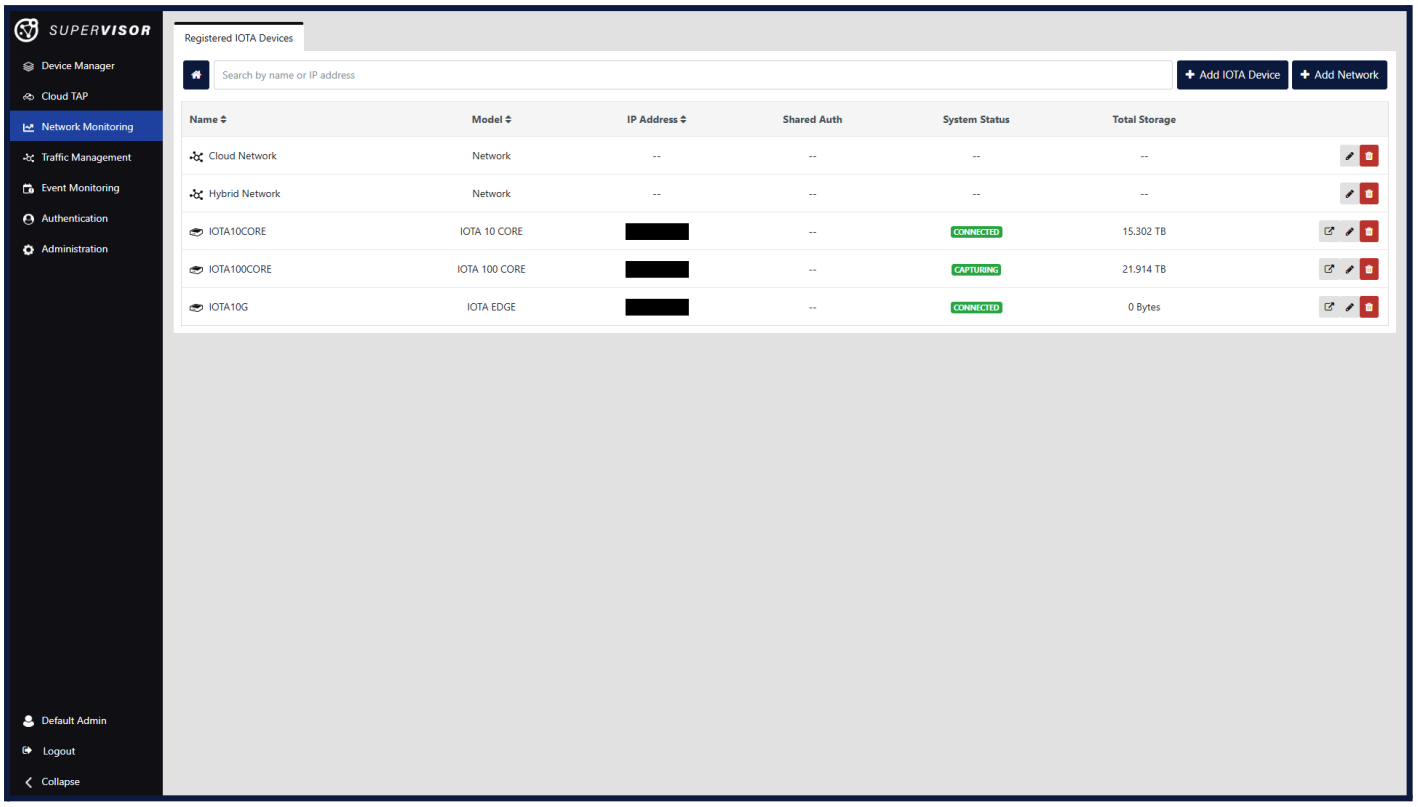
Topology - Kubernetes example



Topology - Azure example

6. Network Monitoring

The **Registered IOTA Devices** tab of the **Network Monitoring** page provides an overview of the IOTA devices managed by the Supervisor, and general information about them, such as their name, model, IP address, shared authentication status, system status, and total storage.



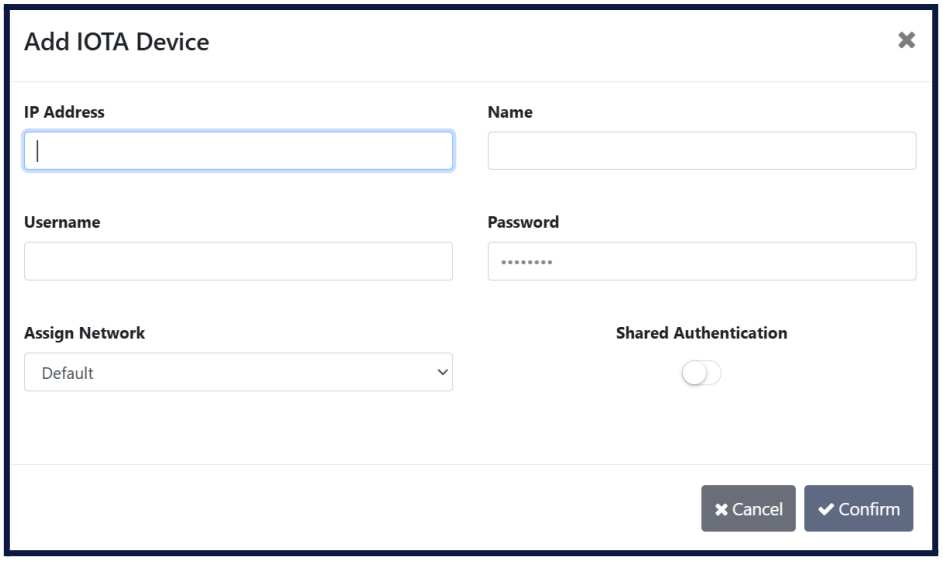
List of registered IOTA devices

From this dashboard, devices and networks can be added, modified, or removed.

Clicking the *Open Device*  button of a device opens this device's management GUI in a new tab.

To add a new device, click the *Add IOTA Device* button in the top right corner of the interface, and enter the device's information in the *Add IOTA Device* window. Enable *Shared Authentication* if you wish to enable Supervisor's centralized authentication function on this device (see Centralized Authentication). The device's information can be changed at a later time by clicking the device's *Edit*  button in the list.

Note: Supervisor centralized authentication is not supported with IOTA 100 CORE, however, IOTA 100 CORE features the same authentication facility as Supervisor, allowing for RADIUS/TACACS+/LDAP authentication.



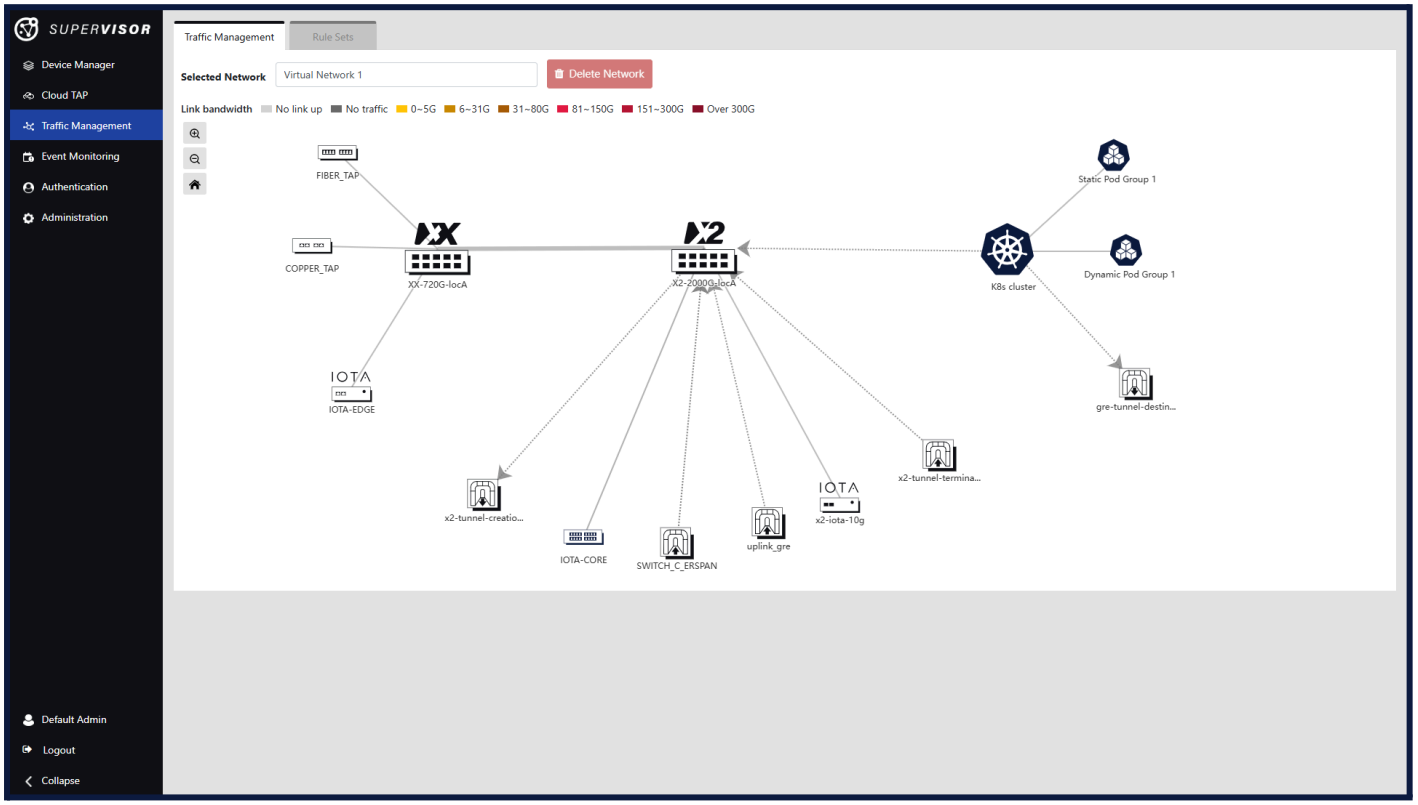
Add IOTA Device window

Note: As of Supervisor v1.2.0, assigning an IOTA device to a network has no effect. Further integration of IOTA in Supervisor will be coming in a future release.

7. Traffic Management

The **Traffic Management** page allows users to define traffic rules operating on an interconnected fleet of Profitap XX-Series and X2-Series packet brokers, Kubernetes clusters, Azure VMs, or a combination. The traffic rules can be used to forward, aggregate and replicate traffic from different devices, K8s pods, and Azure VM interfaces. Supervisor will automatically generate and deploy the necessary device configuration to achieve the desired result.

Prior to creating traffic rules, port groups and packet broker uplinks can be created from the Registered Devices page, and pod groups, interface groups, and tunnel destinations from the Registered Clusters page. This can also be done by clicking a device, K8s cluster, or Azure VM environment in the Traffic Management graphical view.



Traffic Management graphical view

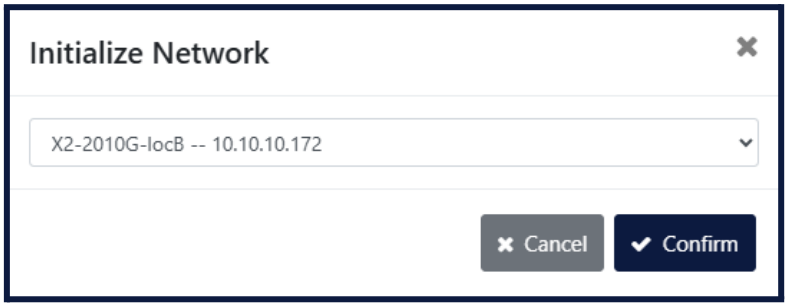
Devices and other entities can be arranged in the graphical view by clicking and dragging them around. The mouse wheel can be used to zoom in and out. While zoomed in, clicking and dragging on an empty space moves the view around.

Clicking on a device or entity in the graphical view opens the related device, K8s cluster, or Azure VM environment details window.

7.1. Network Initialization

A network must first be initialized before using it for traffic management. The network must contain at least one device or virtual environment. To initialize a network, select it in the drop-down menu in the top left corner and click the *Initialize Network* button. In the *Initialize Network* window, select a device or environment, and click the *Confirm* button.

Select a network to initialize

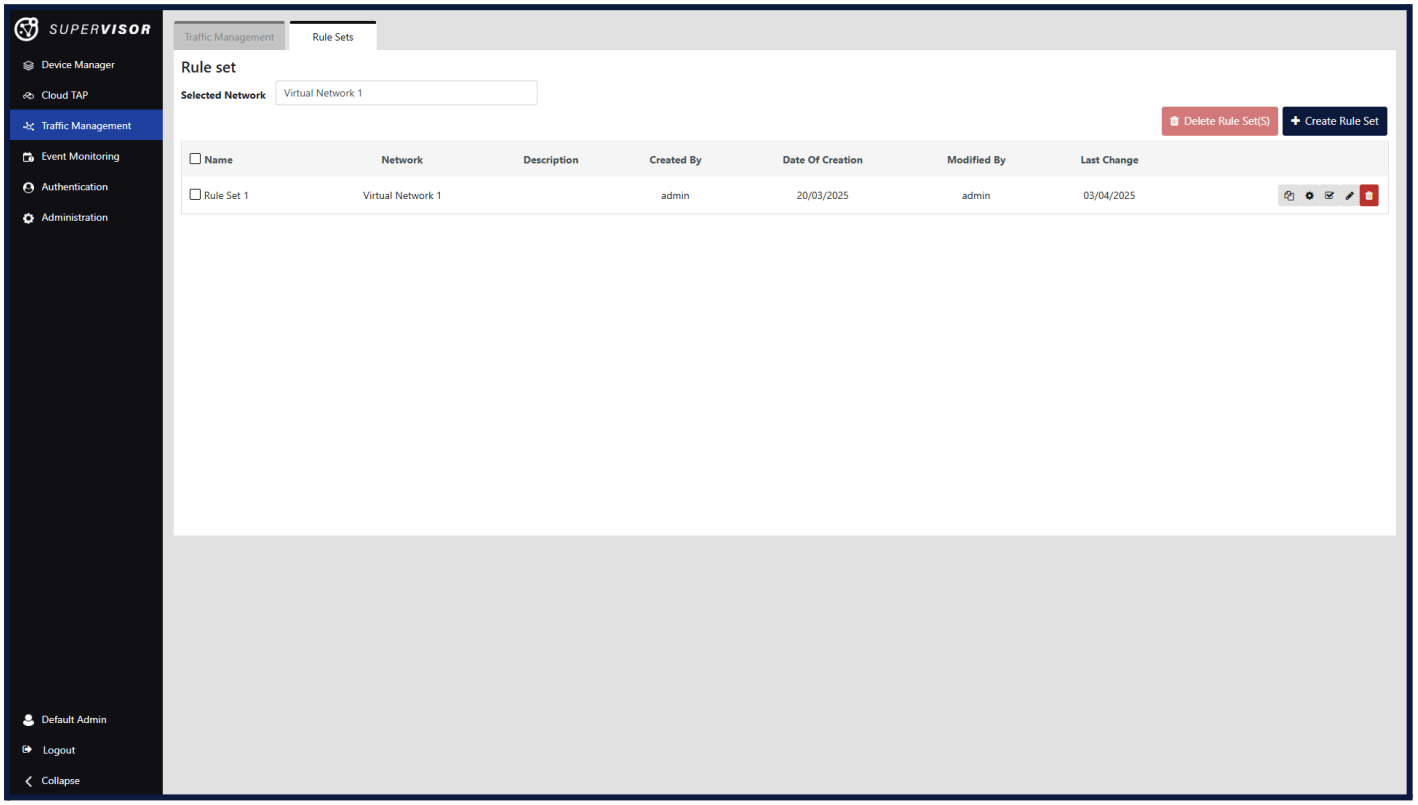


Select a device or environment with which to initialize the network

7.2. Rule Sets

Supervisor uses the registered topology of devices, external devices, K8s pods, and Azure VM interfaces to allow you to perform advanced cross-device traffic management. The configuration of all these elements is covered by the Supervisor rule sets. These are traffic management profiles that can be created, cloned, swapped and modified. Any time a new rule set is applied, the Supervisor system will make sure that the configuration is automatically deployed on the targeted devices.

The **Rule Sets** tab displays the list of existing rule sets for the selected network. Each network operates with its own separate rule sets. Use the *Selected Network* drop-down menu in the top left corner to view and manage the rule sets for a specific network, or select the *All Networks* option for a view of existing rule sets across all networks.



Rule Sets tab displaying the list of rule sets on the selected virtual network

The following actions are available:

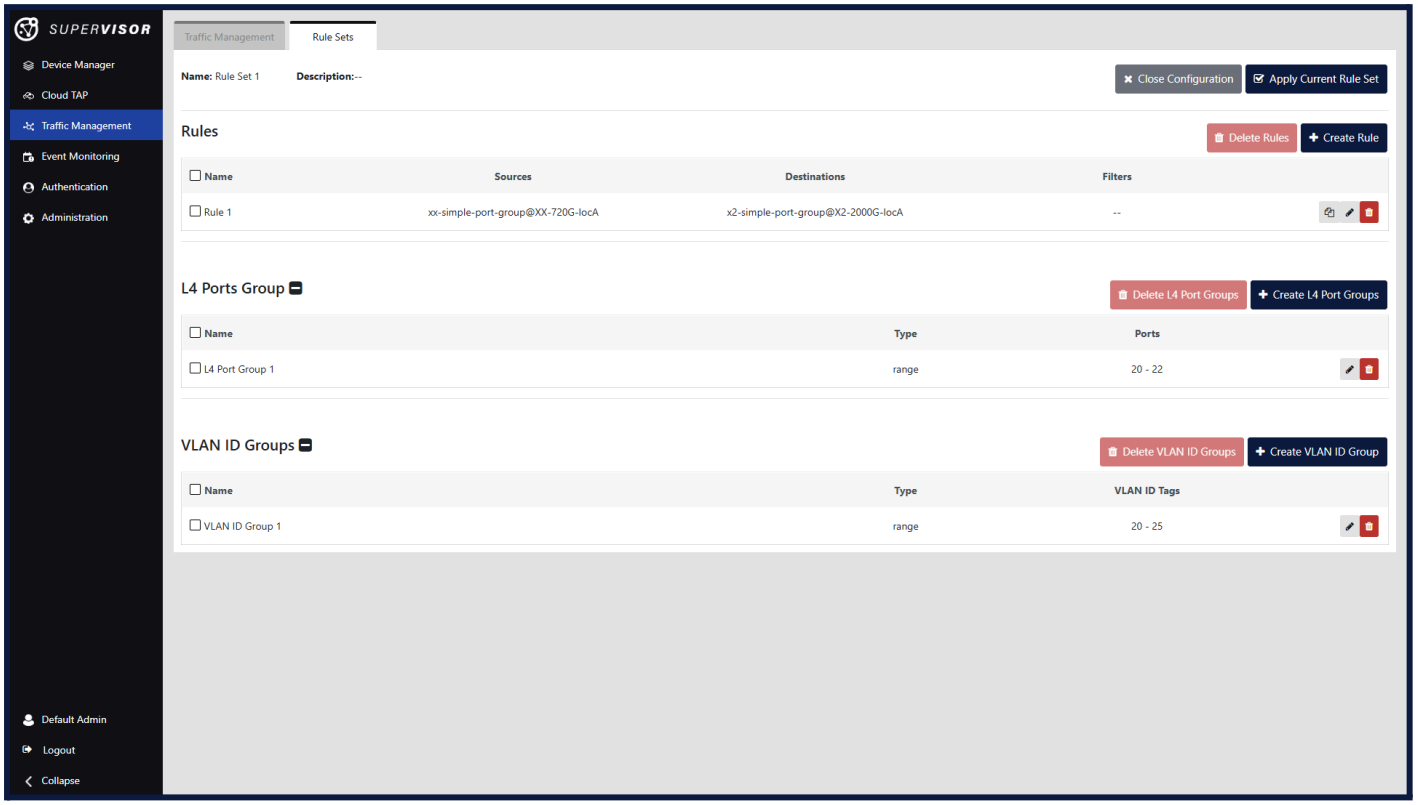
- Create Rule Set Create a rule set
- Clone a rule set
- Configure a rule set
- Apply a rule set
- Rename a rule set
- Delete a rule set

Multiple rule sets can be deleted by selecting one or more rule sets and pressing the *Delete Rule Sets* button.

> **Note:** In order to apply changes to the active rule set, it is necessary to apply the rule set again.

If a rule set is currently active, it is displayed at the bottom of the **Traffic Management** tab. It can be deactivated via the *Deactivate Rule Set* button.

A rule set can contain one or more traffic rules. Clicking the *Configure Rule Set* button displays the configuration page for the selected rule set.



Example of a rule set

From within a rule set, it is possible to manage the traffic rules, and the L4 port groups and VLAN ID groups which can be used in the traffic rule filters. Clicking the *Apply Current Rule Set* button deploys this rule set on the network. Clicking the *Close Configuration* button navigates back to the list of rule sets. Changes to a rule set are saved automatically. Changes to the active rule set are not deployed automatically; the rule set must be applied again in order to propagate the new network configuration.

7.3. Traffic Rules

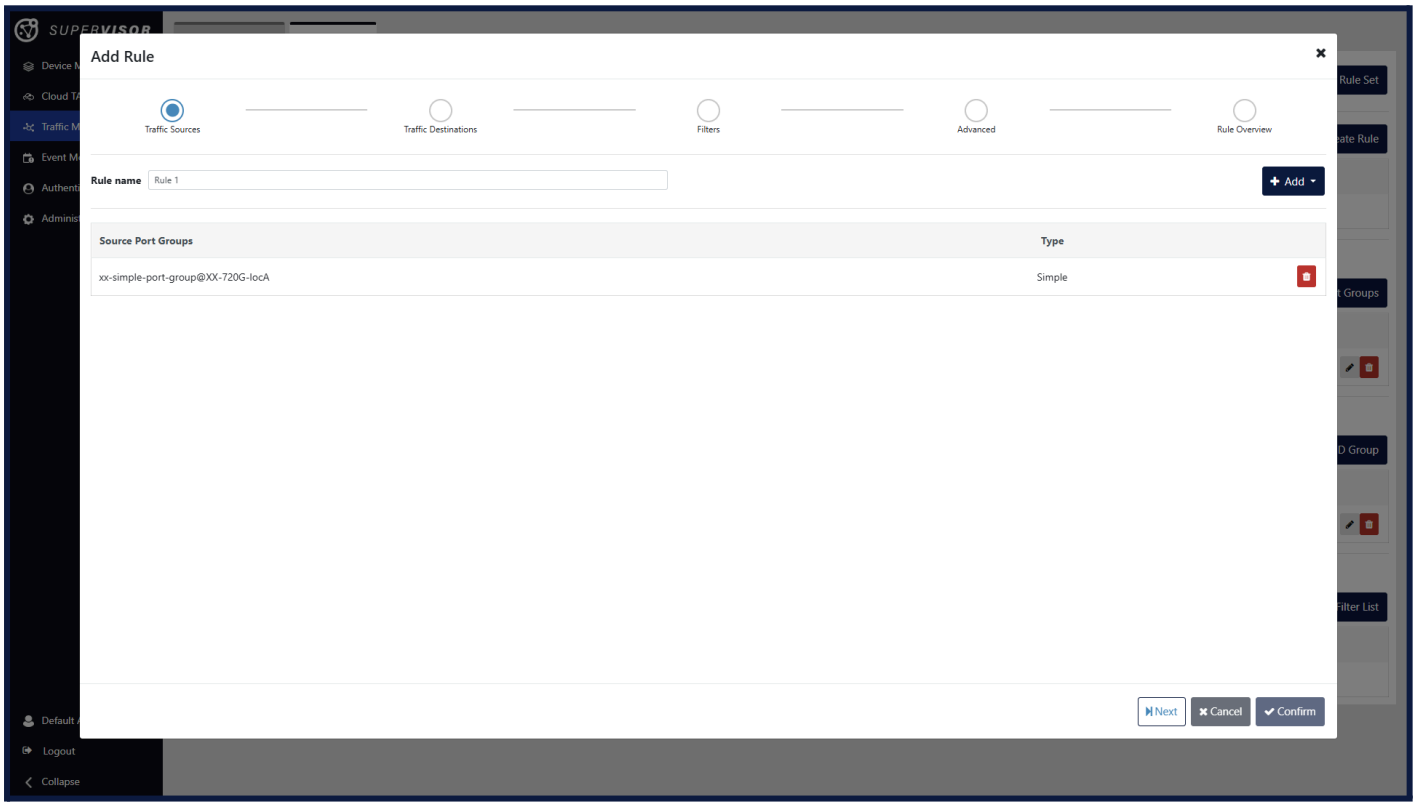
Traffic Rules are at the core definition of the Supervisor traffic management. Each rule allows the definition of the source and destination of the network traffic, as well as filters for that traffic. The rules use the physical uplinks to make sure that the packets reach the intended target.

Click the *Create Rule* button within a rule set to start creating a rule. A basic rule consists of a name, traffic source, and traffic destination. One or more sources and destinations can be defined. Defining more than one source will aggregate the traffic, and defining more than one destination will replicate the traffic. Optionally, filters can be defined if the destinations are port groups on X2-Series devices or K8s/Azure tunnel destinations, and advanced options can be defined if the destinations are port groups on X2-Series devices. If no filter is defined, the rule will allow all traffic. The *Traffic Sources*, *Traffic Destinations*, *Filters*, and *Advanced* tabs are described in the following sections. The *Rule Overview* tab displays a summary of the rule. Click the *Confirm* button to finish creating the rule.

7.3.1. Traffic Sources

Traffic Sources can be simple port groups, external devices or tunnel termination port groups that were created on XX-Series or X2-Series devices, pod groups that were created on K8s clusters, or interface groups that were created on Azure VM environments.

Add a source by clicking the *Add* button, selecting the desired source type, then selecting the source from the drop-down menu, and finally clicking the *Apply Port Group Source* button. Repeat this process to add more sources.



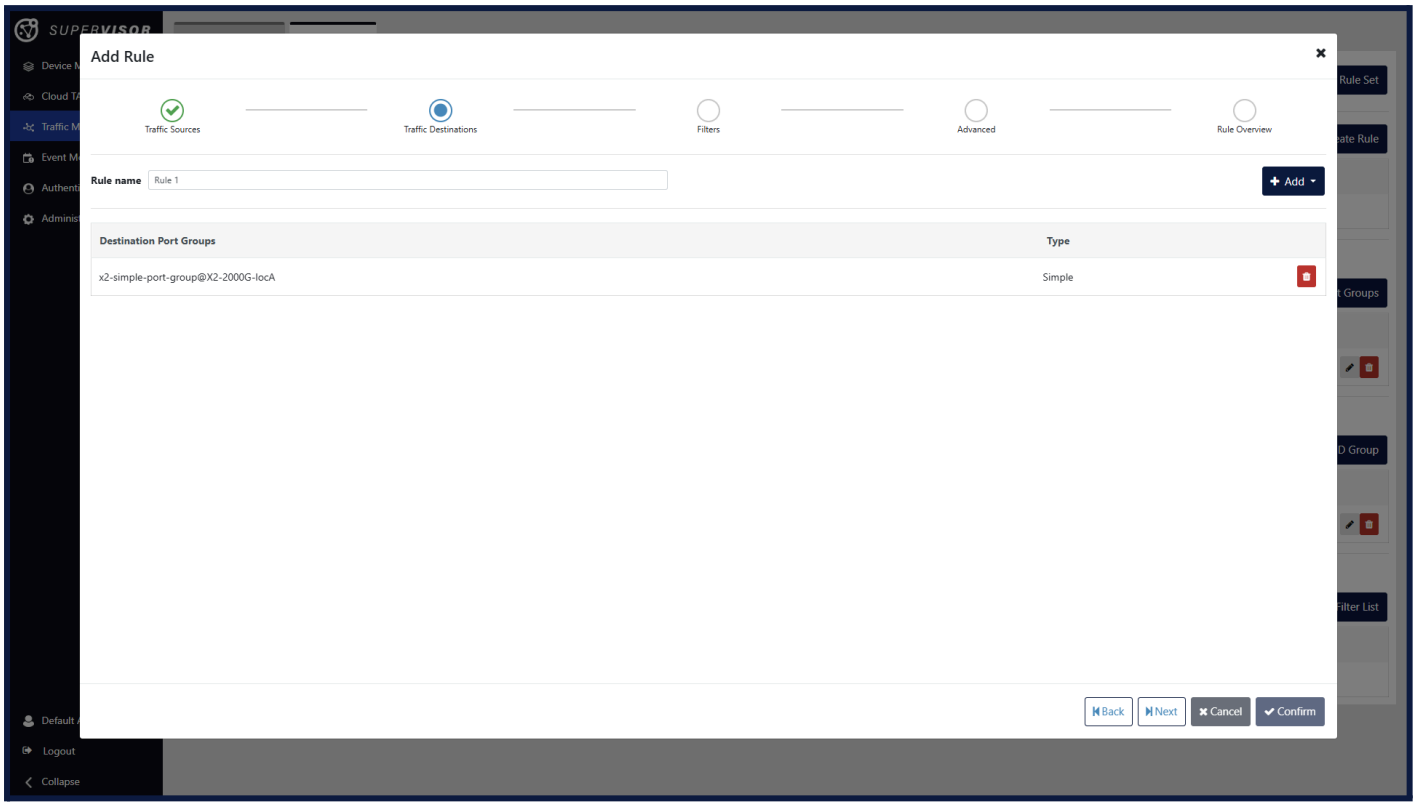
Example of a rule with a simple port group set as traffic source

Click the *Next* button to continue to the *Traffic Destinations* tab.

7.3.2. Traffic Destinations

Traffic Destinations can be simple port groups, external devices or tunnel creation port groups that were created on XX-Series or X2-Series devices.

Add a destination by clicking the *Add* button, selecting the desired destination type, then selecting the destination from the drop-down menu, and finally clicking the *Apply Port Group Destination* button. Repeat this process to add more destinations.

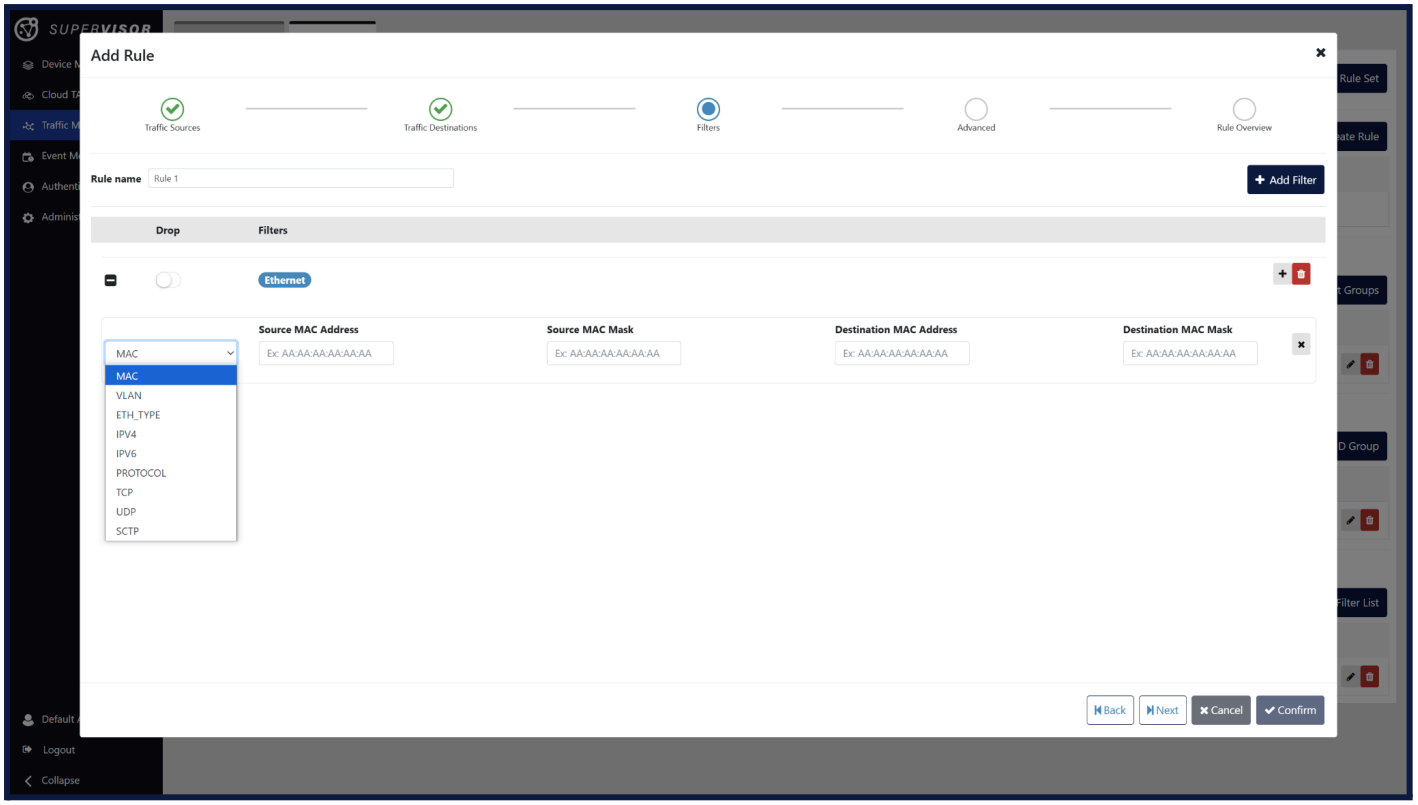


Example of a rule with a simple port group set as traffic destination

Click the *Next* button to continue to the *Filters* tab.

7.3.3. Filters

Filters can be defined if the destinations are port groups on X2-Series devices or K8s/Azure tunnel destinations.



Multiple filters can be created, and each filter can contain one or more statements.

The following actions are available:

-  Add a filter containing one statement
-  Add a statement to a filter
-  Remove a statement from a filter
-  Delete a filter

Each filter can be set as an **allow** filter or a **drop** filter using the *Drop* toggle.

The filter behavior is as follows:

- Filters are logically disjunctive (OR), meaning that any traffic matching any **allow** filter will be allowed through, except for the parts of that traffic that match **drop** filters.
- Any traffic matching any **drop** filter will be dropped.
- If only **allow** filters are set, only the traffic matching these filters will be allowed through.
- If only **drop** filters are set, all traffic will be allowed through, except for traffic that matches any of these **drop** filters.
- If no filters are present, all traffic from the selected sources will be sent to the selected destinations.

- Statements within a filter are logically conjunctive (AND), meaning that each filter only applies to traffic which matches **all** of the statements within that filter.

The **Filters** column of each filter provides an overview of the filter types of all statements present in the filter.

The leftmost drop-down menu of each statement allows the selection of the type of filter for this statement. The rest of the fields and drop-down menus in that statement will depend on the selected filter type.

The available filter types are as follows:

- **MAC:** Filter on source and/or destination MAC address.
- **VLAN:** Filter on VLAN ID. Select a VLAN ID group. VLAN ID groups can be created on the *Rule Sets* page.
- **ETH_TYPE:** Filter on EtherType. Input the hexadecimal EtherType value.
- **IPV4/IPV6:** Filter on source and/or destination IPv4/IPv6 address.
- **PROTOCOL:** Filter on protocol. Input the protocol number.
- **TCP/UDP/SCTP:** Filter on TCP/UDP/SCTP source and/or destination L4 port groups. L4 port groups can be created on the *Rule Sets* page.

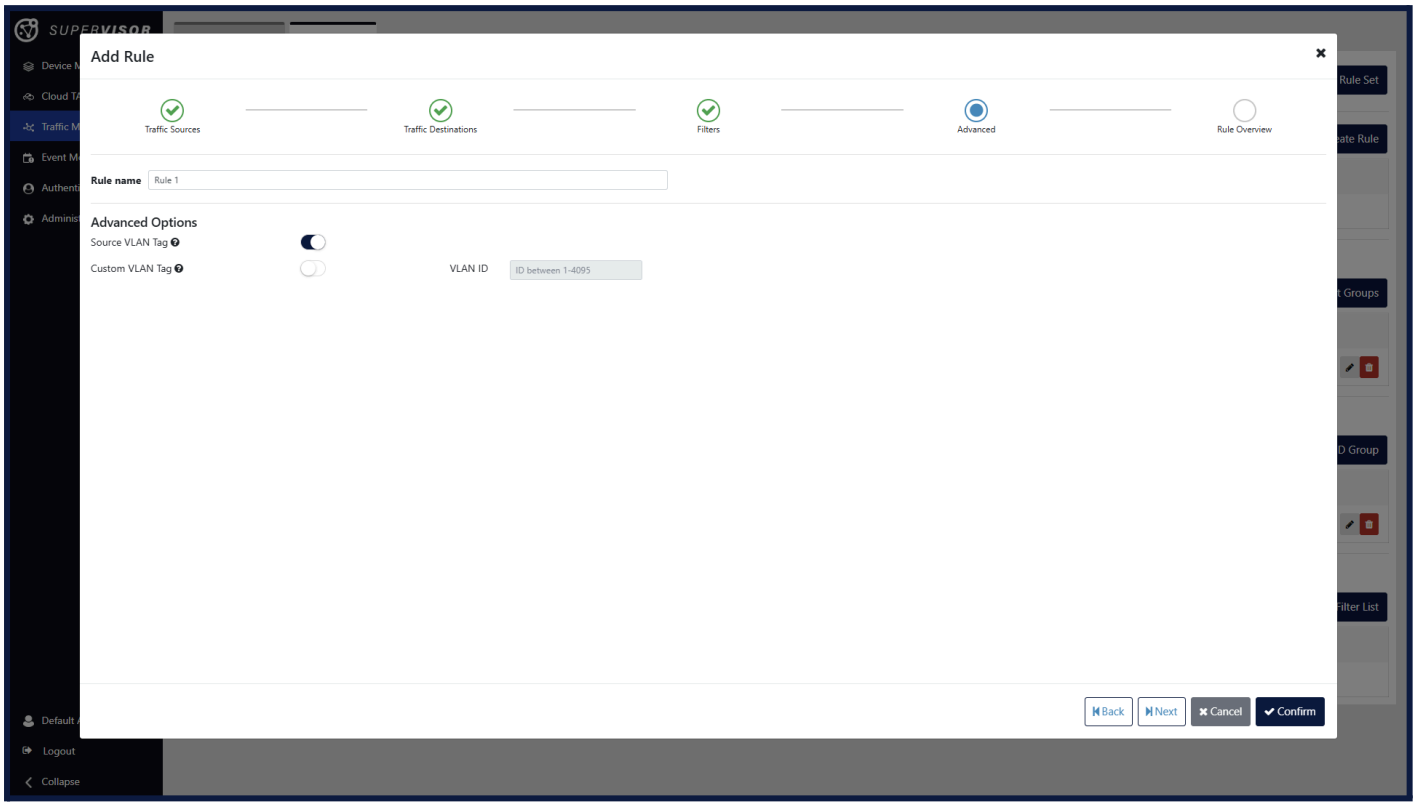
Note: After updating Supervisor to v1.0.0, ICMP/IGMP filter statements will be lost. These can be reconfigured using the *Protocol* statement and the standard protocol number.

7.3.4. Advanced Options

Advanced options can be defined if the destinations are port groups on X2-Series devices.

The available options are as follows:

- **Source VLAN Tag:** Label the outgoing traffic with the VLAN ID defined in the source port group(s) (see External Device Uplink).
- **Custom VLAN Tag:** Label the outgoing traffic with the specified VLAN ID.

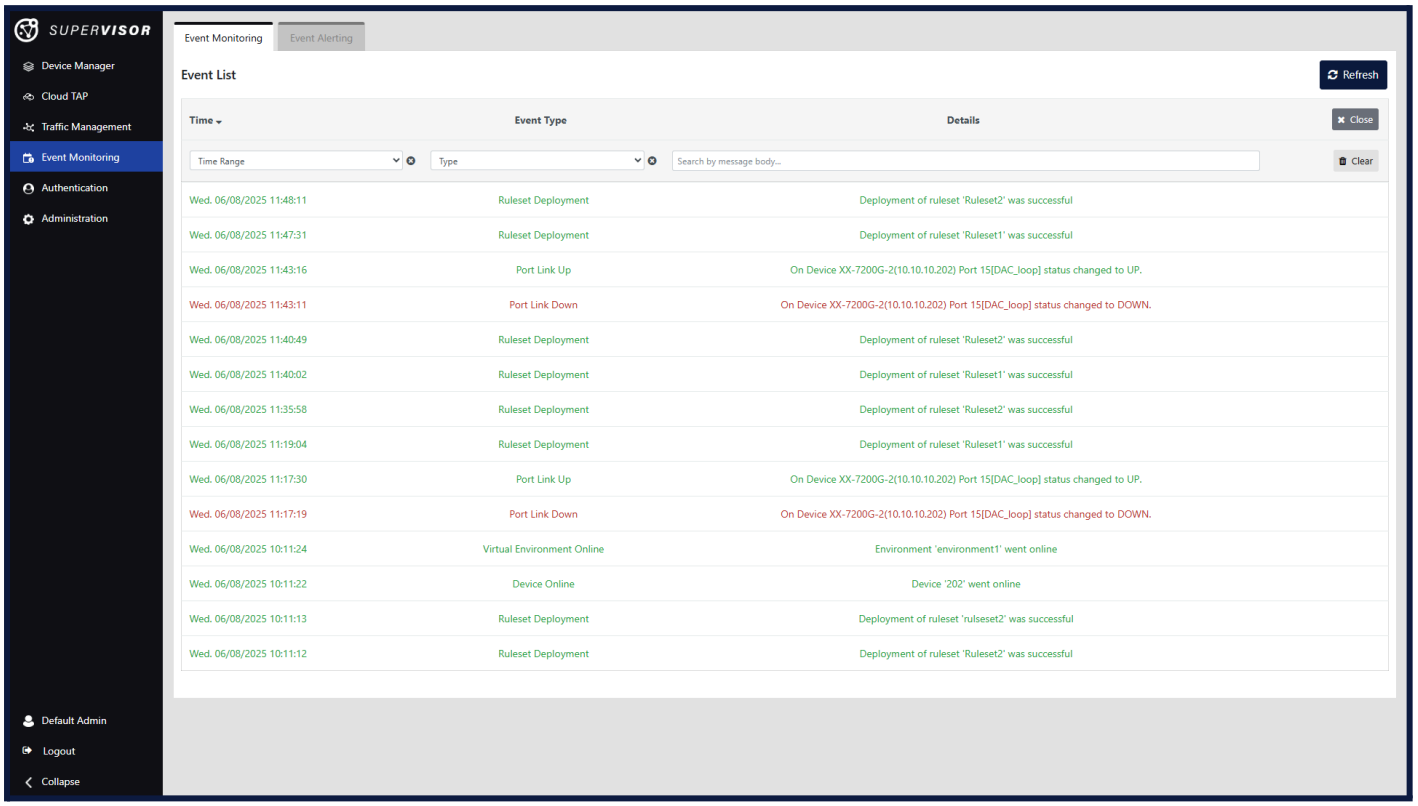


Example of a rule with Source VLAN Tag enabled

8. Event Monitoring

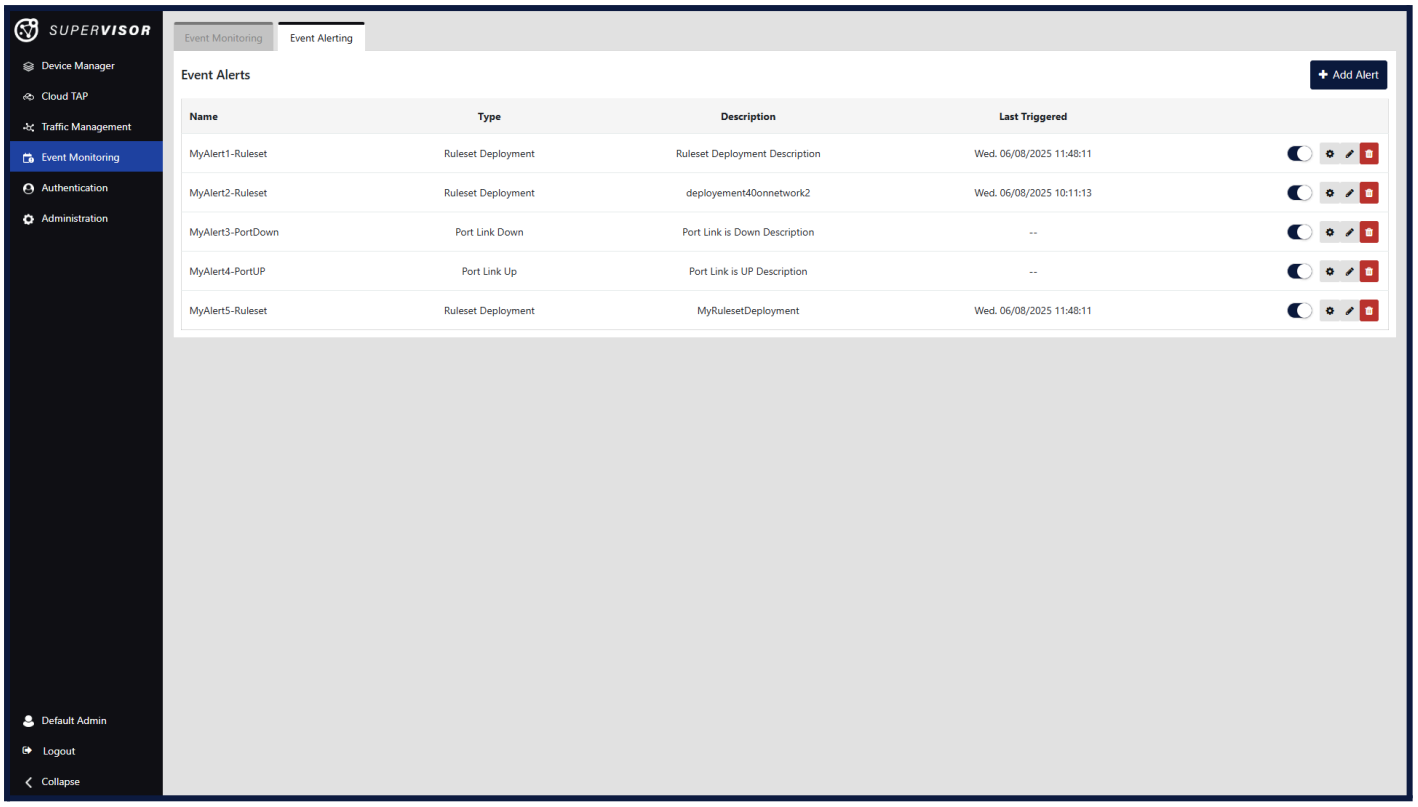
8.1. Event Monitoring

The **Event Monitoring** page displays all of the events detected by the Supervisor. The events can be filtered by time range, event type, and message body, using the *Filter* button and selecting the filtering options. It is also possible to navigate to the device and rule set involved in the event by clicking the event description.



8.2. Event Alerting

The **Event Alerting** page allows you to create email and webhook alerts for certain events.

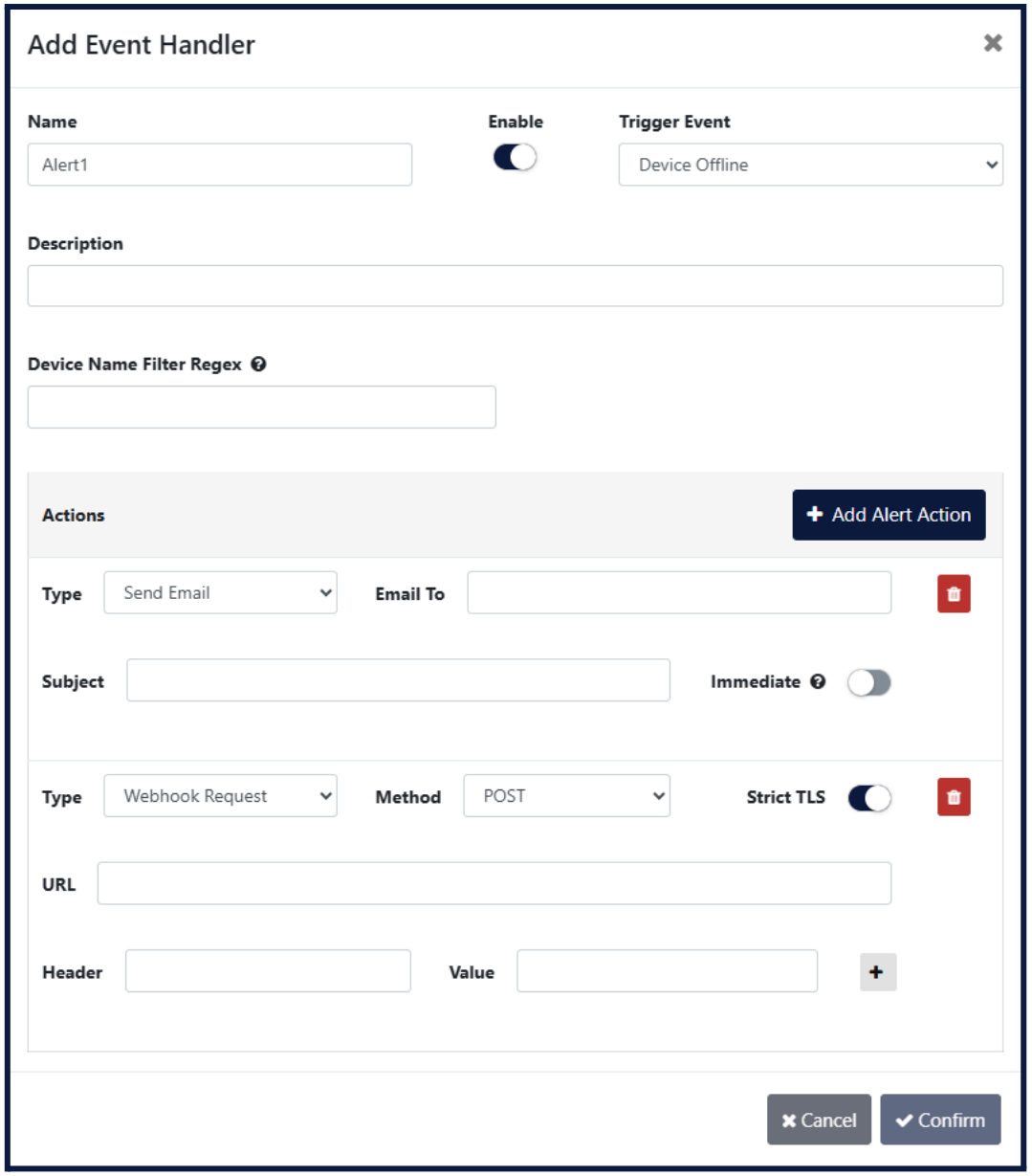


The following actions are available:

-  Create an alert
-  Test an alert
-  Edit an alert
-  Delete an alert

A toggle next to each alert also allows you to enable or disable an alert.

Clicking the *Add Alert* or *Edit Alert* button opens the following window:



- **Name:** A name for this alert.
- **Enable:** Enable or disable the alert.
- **Trigger Event:** Select the type of event triggering this alert.
    - **Ruleset Deployment:** A ruleset is scheduled for deployment.
    - **Ruleset Disabling:** A ruleset is deactivated.
    - **Device Online:** The device transitions to an online state.
    - **Device Offline:** The device transitions to an offline state.
    - **Virtual Environment Online:** The virtual environment transitions to an online state.
    - **Virtual Environment Offline:** The virtual environment transitions to an offline state.
    - **Port Link Up:** The port status changes to UP.
    - **Port Link Down:** The port status changes to DOWN.
    - **Port Stats CRC Error:** CRC errors are detected on the port. If CRC errors cease to increase, no additional events will be generated.
    - **Port Stats Drop:** Frame drops are detected on the port. If frame drops cease to increase, no additional events will be generated.
    - **Port Stats DEDUP Drop:** Duplicate frame drops are detected on the port due to deduplication being enabled. If duplicate frame drops cease to increase, no additional events will be generated.

    - **Port Stats Traffic Increase:** A significant increase in traffic is detected on the port.
    - **Port Stats Traffic Decrease:** A significant decrease in traffic is detected on the port.
    - **Data Usage Warning:** Data usage exceeds 50%, 75%, 90%, and 95%. A separate event will be generated for each threshold that is surpassed.
      Data Quota Usage can be viewed in the License Information section, accessible via Administration > Setup > License Information.
    - **Data Quota Exceeded:** Data quota usage reaches or exceeds 100%.
      Data Quota Usage can be viewed in the License Information section, accessible via Administration > Setup > License Information.
- **Description:** A description for this alert.
- **Network Name Filter Regex:** Filter string to match the network(s) name(s) triggering this alert.
- **Device Name Filter Regex:** Filter string to match the device(s) name(s) triggering this alert.
- **Device Port Filter:** Comma-separated list of filter strings to match the port(s) name(s) or custom label(s) triggering this alert.
- **Cluster Name Filter Regex:** Filter string to match the cluster(s) name(s) triggering this alert.
- **Add Alert Action:** Add an action to execute when the alert is triggered.
- **Type:** Select the action type (Send Email/Webhook Request).
    - **Send Email:**
        - **Email To:** Email address to send the email alert to.
        - **Subject:** Subject email field, used if the *Immediate* option is enabled.
        - **Immediate:** If enabled, the email will be sent as soon as the event occurs. Else, the alert will be queued to be included in the next batch email.
    - **Webhook Request:**
        - **Method:** Specifies the HTTP method to be used for the webhook request.
          Supported values: POST, PUT.
          Use POST to submit data to the endpoint, or PUT to update an existing resource.
        - **Strict TLS:** Enables or disables strict TLS (Transport Layer Security) verification for secure HTTPS connections.
          When enabled, the connection requires a valid, trusted SSL/TLS certificate.
          Recommended: Enable for secure and trusted endpoints.
        - **URL:** Defines the full destination URL to which the webhook request will be sent.
          Must begin with http:// or https:// and point to a valid, reachable endpoint.
        - **Header:** Specifies a custom HTTP header key to be included in the webhook request.
          Common use cases include authorization headers, content type definitions, or custom identifiers.
          Example: Authorization, Content-Type.
        - **Value:** Defines the corresponding value for the specified HTTP header.
          This value is sent along with the header in the webhook request.
          Example: Bearer <token>, application/json.
        - **+ button:** Add a Header/Value row.

Note: SMTP server settings must be configured for email alerts to function (see SMTP Server Configuration).

Appendix A: Alternative Installation Scenarios

A.1. Installing Supervisor on Kubernetes Worker Node as a Pod

The installation can be performed using the following commands in order:

This will install Supervisor on the selected worker node as a pod.

1. Create a directory on the desired **worker node**, to be used to store the supervisor configuration and license:

```
mkdir -p /home/user/supervisor-data/
```

This is only a reference path used for this documentation. If a different path is used, edit the following commands accordingly.

2. Copy the provided license file in the data directory on the **worker node**:

```
cp SFM-010010-10.lic /home/user/supervisor-data/license.lic
```

Replace the name of the file in this command with the actual license file provided.

3. Load the provided Supervisor *docker* container on the **worker node** (replace 'X.Y.Z' with the appropriate version number).

Kubernetes uses containerd's own image store, hence we import into containerd in order for Kubernetes to see it.

```
sudo ctr -n k8s.io images import profitap-supervisor-vX.Y.Z.tar
```

4. Create the following YAML file (e.g. sv.yml) on the **master node**, with image, hostPath path, and nodeSelector hostname adjusted as per your setup:

```
apiVersion: v1
kind: Pod
metadata:
  name: supervisor-pod
  labels:
    app: supervisor
spec:
  hostNetwork: true  # Use host networking
  containers:
    - name: supervisor
      image: profitap-supervisor:v1.1.0  # Use the image name after loading
      imagePullPolicy: IfNotPresent  # Use IfNotPresent since the image is local
#     env:
#       - name: SUPERVISOR_THREADS
#         value: "8"
      volumeMounts:
        - mountPath: /data  # Path in the container
          name: supervisor-storage  # Name of the volume
  volumes:
    - name: supervisor-storage  # Volume name
      hostPath:
        path: /home/profitap/supervisor-data  # Updated local path on the host
  nodeSelector:
    kubernetes.io/hostname: workernode1  # Updated hostname
```

5. Run the sv.yml file with kubectl on the **master node**:

```
kubectl apply -f sv.yml
```

At this point, the Supervisor pod should be running. If you wish to verify that deployment has proceeded correctly, you can check the running containers using the following command:

```
kubectl get pods
```

A.2. Installing Supervisor on Kubernetes Worker Node as a Deployment

The installation can be performed using the following commands in order:

This will install Supervisor on the selected worker node as a deployment.

1. Create a directory on the desired **worker node**, to be used to store the supervisor configuration and license:

```
mkdir -p /home/user/supervisor-data/
```

This is only a reference path used for this documentation. If a different path is used, edit the following commands accordingly.

2. Copy the provided license file in the data directory on the **worker node**:

```
cp SFM-010010-10.lic /home/user/supervisor-data/license.lic
```

Replace the name of the file in this command with the actual license file provided.

3. Load the provided Supervisor *docker* container on the **worker node** (replace 'X.Y.Z' with the appropriate version number).

Kubernetes uses containerd's own image store, hence we import into containerd in order for Kubernetes to see it.

```
sudo ctr -n k8s.io images import profitap-supervisor-vX.Y.Z.tar
```

4. Create the following YAML file (e.g. sv-deployment.yml) on the **master node**, with image, hostPath path, and nodeSelector hostname adjusted as per your setup:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: supervisor-deployment
  labels:
    app: supervisor
spec:
  replicas: 1
  selector:
    matchLabels:
      app: supervisor
  template:
    metadata:
      labels:
        app: supervisor
    spec:
      hostNetwork: true  # Use host networking
      containers:
        - name: supervisor
          image: profitap-supervisor:v1.1.0
          imagePullPolicy: IfNotPresent  # Use local image if available
#         env:
#           - name: SUPERVISOR_THREADS
#             value: "8"
          volumeMounts:
            - mountPath: /data
              name: supervisor-storage
      volumes:
        - name: supervisor-storage
          hostPath:
            path: /home/profitap/supervisor-data
      nodeSelector:
        kubernetes.io/hostname: workernode1
```

5. Run the sv-deployment.yml file with kubectl on the **master node**:

```
kubectl apply -f sv-deployment.yml
```

At this point, the Supervisor deployment should be running. If you wish to verify that deployment has proceeded correctly, you can check the running containers using the following commands:

```
kubectl get pods
kubectl get deployments
```

A.3. Installing Supervisor on Kubernetes Worker Node as a Deployment with Remote Data Directory

The installation can be performed using the following commands in order:

This will install Supervisor on any worker node as a deployment with remote data directory. The remote data directory is an NFS share (CIFS is not supported).

1. Create the following YAML file (e.g. pv-nfs.yml), with server IP and path adjusted as per your setup:

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: nfs-pv
spec:
  capacity:
    storage: 5Gi
  accessModes:
    - ReadWriteMany
  nfs:
    path: /volume1/remotedata
    server: 10.10.10.233
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs-storage
```

Run the pv-nfs.yml file with kubectl:

```
kubectl apply -f pv-nfs.yml
```

This will create a remote persistent volume.

Note: NFS tools need to exist on worker nodes. Tools can be installed using:

```
sudo apt update
sudo apt install -y nfs-common
```

2. Create the following YAML file (e.g. pvc-nfs.yml):

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: supervisor-pvc
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 5Gi
  storageClassName: nfs-storage
  volumeName: nfs-pv # Explicitly bind to the specific PV
```

Run the pvc-nfs.yml file with kubectl:

```
kubectl apply -f pvc-nfs.yml
```

This will create the PVC on kubernetes cluster.

3. Create the following YAML file (e.g. sv-deployment-remote.yml), replacing the image name with the appropriate image name:

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: supervisor-deployment
  labels:
    app: supervisor
spec:
  replicas: 1
  selector:
    matchLabels:
      app: supervisor
  template:
    metadata:
      labels:
        app: supervisor
    spec:
      hostNetwork: true  # Use host networking
      containers:
        - name: supervisor
          image: profitap-supervisor:v1.1.0
          imagePullPolicy: IfNotPresent  # Use local image if available
          # env:
          #   - name: SUPERVISOR_THREADS
          #     value: "8"
          volumeMounts:
            - name: supervisor-storage
              mountPath: /data  # Same container path
      volumes:
        - name: supervisor-storage
          persistentVolumeClaim:
            claimName: supervisor-pvc  # Using the PVC instead of hostPath
```

Run the sv-deployment-remote.yml file with kubectl:

```
kubectl apply -f sv-deployment-remote.yml
```

At this point, the Supervisor deployment should be running. If you wish to verify that deployment has proceeded correctly, you can check the running containers using the following commands:

```
kubectl get pods
kubectl get deployments
```

Legal

Disclaimer

The manufacturer makes no representations or warranties with respect to the contents hereof and specifically disclaims any implied warranty of merchantability or fitness for any particular purpose. The manufacturer reserves the right to revise this publication and to make changes in the content thereof without obligation of the manufacturer to notify any person of such revision or changes.

Copyright

Trademarks

The trademarks mentioned in this manual are the sole property of their owners.

Profitap HQ B.V.
High Tech Campus 84
5656 AG Eindhoven
The Netherlands
sales@profitap.com
www.profitap.com