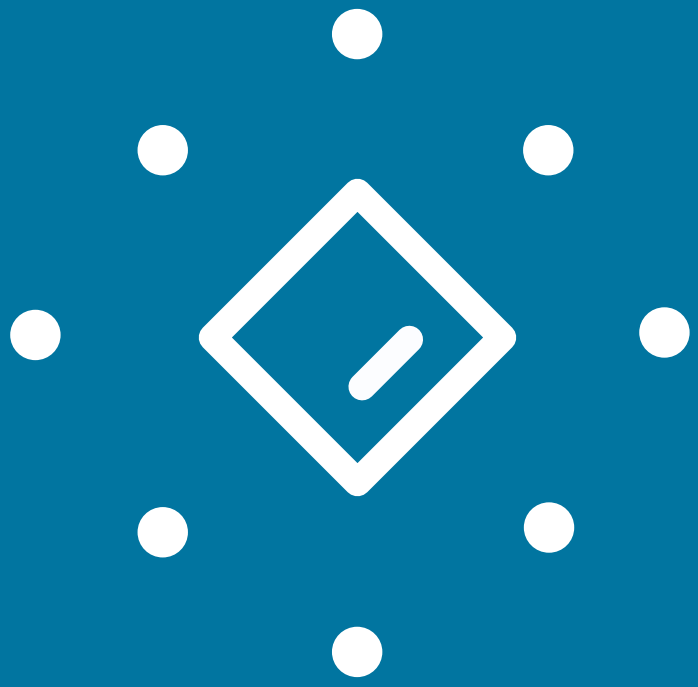
NOZOMI NETWORKS



Remote Collector
Installation Guide

N2OS v26.6.0 — 2026-11-13

Legal notices

Information about the Nozomi Networks copyright and use of third-party software in the Nozomi Networks product suite.

Copyright

Copyright © 2013-2026, Nozomi Networks. All rights reserved. Nozomi Networks believes the information it furnishes to be accurate and reliable. However, Nozomi Networks assumes no responsibility for the use of this information, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright, or other intellectual property right of Nozomi Networks except as specifically described by applicable user licenses. Nozomi Networks reserves the right to change specifications at any time without notice.

Third Party Software

Nozomi Networks uses third-party software, the usage of which is governed by the applicable license agreements from each of the software vendors. Additional details about used third-party software can be found at https://security.nozominetworks.com/licenses.

Contents

Chapter 1. Introduction 5
Remote Collector overview 7
Guardian with Remote Collectors 8
Installation options 11
Updates 11
Chapter 2. Physical installation 13
Physical installation 15
Install a physical sensor 16
Chapter 3. Virtual installation 19
Virtual installation 21
Virtual machine sizing 22
Check the CPU of a virtual machine 23
Check the RAM of a virtual machine 24
Install a virtual machine 25
Expand a disk on a virtual machine 27
Add a secondary disk to a virtual machine 28
Add a monitoring interface to the virtual machine 29
Chapter 4. Container installation 31
Docker installation 33
Docker prerequisites 34
Build a Docker image 35
Install a sensor in a Docker container 36
Update a Docker container 37
Remote Collector container installation on the Cisco Catalyst 9300 38
Prepare the Remote Collector container as a IOx packet 40
Configure the Remote Collector container on the Cisco Catalyst 9300 43
Chapter 5. Basic configuration 45
Do the basic configuration 47
Chapter 6. Additional settings 51
Network requirements 53
Operator access to Guardian, CMC, and RC 53
Communications between Guardian, CMC, and RC 53
Operator access to Vantage 54
Communications between Guardian, CMC, and Vantage 54
Install a CA certificate 55
Configure certificate rotation 56
Enable SNMP 57

Internal firewall configuration 60
Management-interface packet-rate protection 61
Disable internal packet-rate protection of the management interface 61
Configure IPv6 62
Enable 802.1x on the management interface 63
USB port disablement 68
Chapter 7. Deployment 69
Connect a Remote Collector to Guardian 71
Configure the Remote Collector 73
Set the time zone 74
Enable bandwidth throttling 75
Enable multiplexing 77
Configure the site name and description 78
Set the compression strategy from a shell console 79
Set the compression strategy from Guardian or CMC 80
Enable traffic forwarding 82
Configure custom certificates 85
Disable a Remote Collector 86
Chapter 8. Compatibility Reference 87
Configuring SSH profiles 89
SSH compatibility 90
HTTPS compatibility 93
Supported SSH protocols in FIPS mode 94
Chapter 9. Federal Information Processing Standards 95
Federal Information Processing Standards 97
FIPS mode status 98
Enable FIPS mode 99
Disable FIPS mode 101
Glossary 103

Chapter 1. Introduction



Remote Collector overview

Remote Collectors let you deploy sensors in multiple isolated locations. Remote Collectors must be connected to a Guardian and act as a remote interface, that broaden its capture capability.

Remote Collectors are low-resource sensors that capture data from distributed locations and send it to Guardian(s) for further analysis. A Remote Collector is typically installed in isolated areas, such as windmills, or solar power fields, where it monitors multiple small sites. Traffic is encrypted. The Remote Collector firmware receives automatic updates from the connected Guardian.

The relationship between a Remote Collector and a Guardian is similar to that between a Guardian and a *Central Management Console (CMC)*, but with some key differences. A Remote Collector:

- Does not process sniffed traffic, it just forwards it to the Guardian to which it is attached
- Has no *graphical user interface (GUI)*
- A Remote Collector has bandwidth limitations

You must enable a Guardian to receive traffic from a Remote Collector. Once it has been enabled in the Guardian, the Remote Collector provides an additional (virtual) network interface, called a **remote-collector** that aggregates the traffic of all the Remote Collectors connected to it. You can open the Guardian's **Sensors** page to inspect all the Remote Collectors that are currently connected.

Each Remote Collector forwards its sniffed traffic to a set of Guardians. Multiple Remote Collectors can connect to a Guardian. To avoid third-party interception, traffic is encrypted with high security measures over the *transport layer security (TLS)* channel.

Certificate-based authentication

The channel is established with a mutual authentication scheme that validates dedicated X.509 certificates on both sides:

- The Guardian checks that the Remote Collector certificate is trusted.
- The Remote Collector checks that the Guardian certificate is trusted.

These certificates are separate from the certificate used by Guardian for the *hypertext transfer protocol secure (HTTPS)* endpoint.

By default, sensors generate their own certificates (one-year validity, self-signed) and periodically perform a rotation procedure to ensure that communication does not break due to expired certificates. For more details, see Configure certificate rotation (on page 56).

You can provision your own certificates instead, in which case the periodic rotation is not performed. For more details, see Configure custom certificates (on page 85).

Regardless of their origin, traffic starts flowing between the Remote Collector and its upstream Guardians only after the certificates are successfully exchanged and trusted via the *HTTPS* control channel.

Guardian with Remote Collectors

A description of how Remote Collectors and Guardians work together.

General

Guardian lets you monitor the health of all the Remote Collectors that are connected to it.

In Guardian, you can select the **Sensors** page to inspect the health of the Remote Collectors. When you select a Remote Collector, an information pane shows on the right with detailed information. This includes the health status of the Remote Collector, and the timestamp of the last received payload traffic.

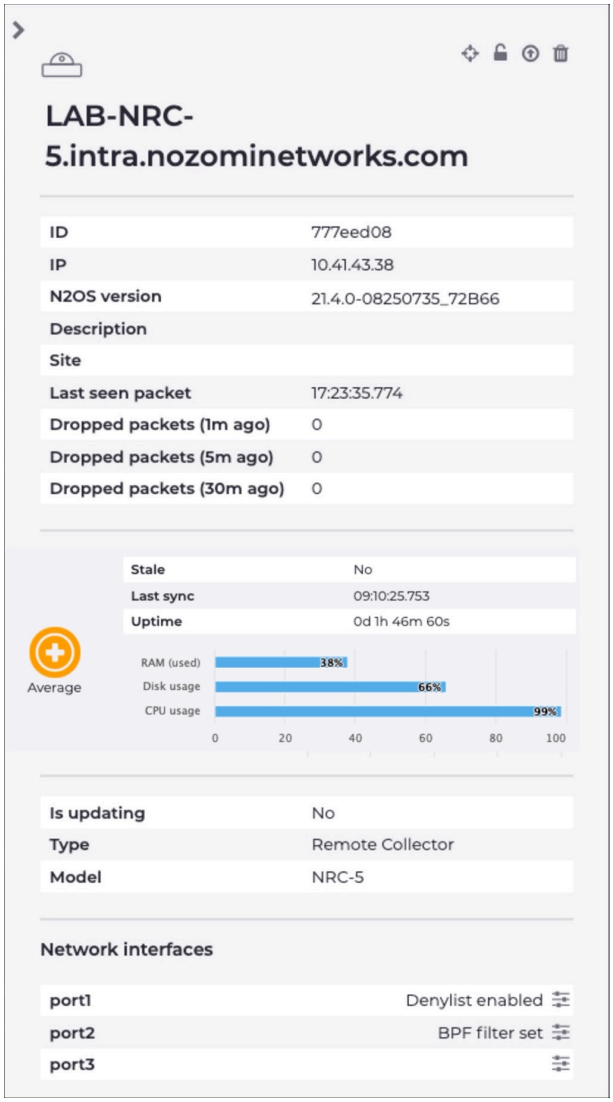


Figure 1. Information pane

At the bottom of the pane, a list of Remote Collector network interfaces is shown. For each network interface, there is a **Configure** button that lets you:

- Upload a denylist
- Enable a denylist
- Disable a denylist

It also lets you set, or unset, *Berkeley Packet Filter (BPF)* in the same way as for the Guardian network interfaces.

Guardian system health

Guardian's system health is communicated through qualitative strings. The possible values for system health are:

- `Unreachable`
- `Poor`
- `Average`
- `Good`

The system health is a weighted average of:

- *random-access memory (RAM)*: with a weight of 0.35
- *central processing unit (CPU)*: with a weight of 0.15
- Disk: with a weight of 0.50

The `unreachable` status indicates a sensor that has not reached out to the Guardian for a long time and is considered stale. The other health levels (poor, average and good) are determined based on resource usage.

If all of the values of *RAM*, disk, or *CPU* usage are less than 80%, the status is `good`.

If at least one of these values is over 80%, the below formula is used to calculate the status:

Average: {*RAM*: 38% * 0.35 + Disk: 66% * 0.5 + *CPU*: 99% * 0.15} = 61%

100%-61% = 39%

If the result is less than 30, the status is `Poor`

If the result is greater than 30, but less than 80, the status is `Average`

If the result is greater than 80, the status is `Good`

Packet origins

The origin of the packets is tracked internally by Guardian and is displayed in several locations, such as in the Nodes tab of the Network page.

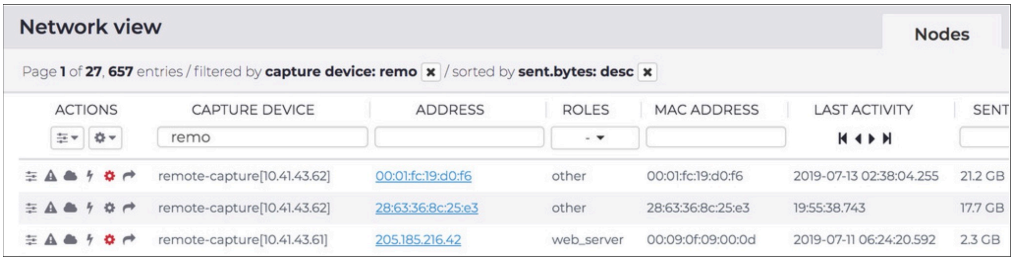


Figure 2. Network view - Nodes tab

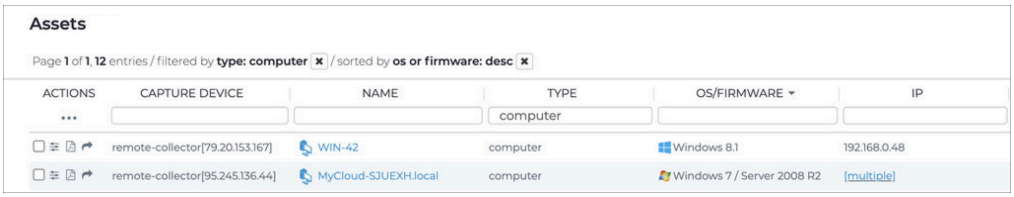


Figure 3. Assets page

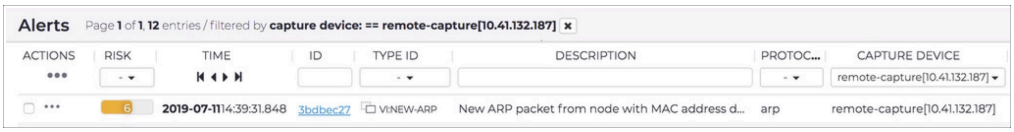


Figure 4. Alerts page

Installation options

A list of the different installation options for Nozomi sensors.

You can install Nozomi Networks sensors in these different methods:

- A physical installation
- A virtual installation
- A Docker installation

Updates

Information on how Remote Collectors are updated.

A Remote Collector's release update depends on the Guardian that it is attached to.

Remote Collectors receive automatic updates from the Guardian to which they are attached. The Remote Collector updates to the Guardian version, only if the current Remote Collector firmware version is older than that of the Guardian.

Note:
The Remote Collector container does not update automatically.

A Remote Collector has no graphical interface. The only other method for changing the version of a Remote Collector is to use the manual procedure described in **Software update and rollback**, in the **Maintenance Guide**.



Chapter 2. Physical installation



Physical installation

A physical installation is a deployment of a dedicated physical hardware sensor that will monitor your network. Typically, Nozomi Networks supplies these physical sensors.

Physical installations are best-suited for organizations that:

- Do not use virtualized architectures
- Manage highly-distributed, dense networks
- Have special environmental conditions

Install a physical sensor

Before you can use your Nozomi Networks physical sensor in your network, you must install it.

About this task

If you purchased a physical sensor from Nozomi Networks, the appropriate release of the *Nozomi Networks Operating System (N2OS)* is already installed on it. If your organization uses a different release of the *N2OS*, then see **Software updates and rollbacks** in the **Maintenance Guide** to upgrade the release.

Procedure

1. Follow the procedure described in the **Quick Start Guide** that you received with the sensor.

   Note:

   The status of your sensor should now:

   - Be installed in a rack
   - Be powered on
   - Have its monitoring ports connected to the relevant network switches

2. Connect the appropriate null-modem serial cable to the sensor's serial console.

   Table 1. Null-modem serial cables

Physical sensor model	Connector type
NRC5 Remote Collector	DB9 (female)

3. Open a terminal emulator.

   Note:

   Use the appropriate terminal emulator for your *operating system (OS)*:

   - On Windows, **Hyper Terminal** or **Putty**
   - On macOS or other *nix platforms, **cu** or **Minicom**

4. Connect and set the speed to 9600 bauds with no parity bit set.

> **Note:**
> Starting with *N2OS* 24.6.0, the **NS1R-RS1R** Guardian sensor typically uses 115200 baud with no parity. In some cases, however, it may require 9600 baud. If you're unsure, test both settings.

5. **Optional:** Connect with *secure shell (SSH)*. Use the default network settings:
   IP address: 192.168.1.254
   Netmask: 255.255.255.0
   GW: 192.168.1.1

   Result: A login prompt for the sensor shows.

6. Do the **Basic configuration**.

> **Important:**
> When you perform the basic configuration, make sure that you enter a valid default gateway. Without a default gateway, the system might not be able to reach your update servers, *CMC*, or external hosts required for full functionality.



Chapter 3. Virtual installation



Virtual installation

Before you install a virtual machine (VM), you should make sure that you are familiar with the minimum requirements, and the supported formats and software that you can use with the Nozomi Networks software products.

Virtual installations are best-suited for organizations that:

- Need to separate the lifecycle of hardware from software
- Are looking to centralize the management of their assets
- Value the ability to easily scale their solutions

The Nozomi Networks solution is available in these formats:

- *open virtual appliance (OVA)*
- *virtual hard disk (VHD)*

When you deploy the *virtual machine (VM)*, use a hypervisor that supports *OVA* or *VHD* file formats, such as:

- HyperV
- KVM
- Proxmox
- VMware
- XEN

All components should be in good working condition. The overall hypervisor load must be under control, with no regular ballooning, so as to avoid unexpected behavior, such as dropped packets or overall poor system performance.



Important:

After deploying the virtual appliance and starting the *VM*, you must access the console and perform the basic configuration. During this process, it is **mandatory** to enter a valid default gateway. Without a default gateway, the system may be unable to reach external services such as the *CMC*, update servers, or other network resources.

Virtual machine sizing

A description of the virtual machine (VM) sizing requirements for Nozomi Networks sensors.

The following information describes the minimum size requirements for sensor instances when they run on virtual machines. These values are based on assumed amounts of nodes and throughput, and are based on a simplified model. You should consider these recommendations as a starting point for calculating the best size for your *VM*.

There are more elements affecting the instantaneous and average use of resources, such as the:

- Hypervisor hardware
- Specific protocol distributions
- Loading time machine snapshots
- Running of queries on big data sets etc.

Because of this, different settings might be required and should be tested during deployment.
In the table below, network elements are defined as the sum of nodes, links, and variables.

Check the CPU of a virtual machine

The amount of central processing units (CPU) from the hypervisor might not correspond with the actual amount seen from within the virtual machine (VM). Therefore, it is important to confirm that your VM has sufficient CPUs to support your configuration.

About this task

Important:
The *CPU* resources must be configured in accordance with the intended usage. Before deployment, verify your sizing requirements (on page 22) in this documentation.

Procedure

1. Open a terminal.
2. Enter this command:

   ```
   sysctl hw.ncpu
   ```

   Result: The terminal will show the number of *CPU* cores allocated to the *VM*.
3. If the number of allocated *CPUs* is insufficient for your configuration, you need to identify the correct amount.
   The recommended resource allocation for a Remote Collector is four *CPUs*.
4. To confirm that the *RAM* allocation in the hypervisor matches your sizing requirements for the intended deployment, see Check the RAM of a virtual machine (on page 24)

Check the RAM of a virtual machine

The amount of random-access memory (RAM) from the hypervisor might not correspond with the actual amount seen from within the virtual machine (VM). Therefore, it is important to confirm that your VM has sufficient RAM to support your configuration.

About this task

Important:
RAM and *CPU* resources must be configured in accordance with the intended usage. Before deployment, verify your sizing requirements (on page 22) in this documentation.

Procedure

1. Open a terminal.
2. Enter this command:

   ```
   sysctl hw.physmem
   ```

   Result: The terminal will show the physical memory of the *VM*.
3. To confirm that the *CPU* allocation in the hypervisor matches your sizing requirements for the intended deployment, see Check the CPU of a virtual machine (on page 23).

Install a virtual machine

Install the Remote Collector on a virtual machine configured for N2OS. Follow the hypervisor setup guidelines and disk compatibility requirements to ensure successful deployment. This process prepares the VM for secure traffic collection and remote monitoring.

Before you begin

Make sure that:

- The *VM OS* is set to FreeBSD 12 or later versions (64-bit)
- The *VM* compatibility version is set to 22+, such as ESXi 9.0 or later (for example, VM version 22), if possible; however, this depends on the vSphere version level
- If Secure Boot is available in the hypervisor, disable it
- If Dynamic Memory is available in the hypervisor, disable it
- The disk is created using *thick provisioning*, not *thin provisioning*. Its size must be fixed, not dynamic

About this task

To install the *VM* in the hypervisor, you need to configure the *VM* to enable external access. If you are not familiar with how to import the *OVA VM* in your hypervisor environment, refer to your hypervisor's manual or contact your hypervisor's support service.

These disk storage types are recommended and supported:

- LSI Logic
- LSI Logic SAS
- SATA

Important:
N2OS does not support VMware Paravirtual SCSI (PVSCSI) storage type.

Important:
IDE disk storage is not supported when deploying *N2OS* on VMware hypervisors. Use SCSI or SATA instead.

Procedure

1. Make sure that you are familiar with the minimum resource requirements.
2. Import the *VM* in the hypervisor and configure the resources.
3. Wait for the *VM* to finish importing.
4. In the hypervisor settings for the *VM* disk, set the desired size.

Note:
Some hypervisors, such as VMware ESX >= 6.0, permit you to change the disk size at this stage.

5. If your hypervisor does not permit you to change the disk size at this stage, do not continue with this procedure until you have completed the procedure: Add a secondary disk to a virtual machine (on page 28).
6. Boot the *VM*.

   Result: The *VM* boots into a valid *N2OS* environment.
7. Log in as admin.

   Result: You are logged in instantly as no password is set by default.
8. To go to privileged mode, enter this command:

   ```
   enable-me
   ```

   Result: You can now perform system changes.

Expand a disk on a virtual machine

The monitoring scope of your sensor will dictate the size requirements of your virtual hard drive. Do this procedure to adjust the size of the virtual hard drive.

Before you begin

Edit your *VM* settings from the hypervisor.

Important:
The *OVA* and VHDX images are built with minimum disk size and are not suitable for production deployments. If your hypervisor supports it, expand the disk before you start the *VM*. For more details see Virtual machine sizing (on page 22).

Procedure

1. From the hypervisor settings, expand the *VM* disk to the correct size for your monitoring scope.
2. Restart the *VM*.
3. Log in to the *VM*.
4. To go to privileged mode, enter this command:

   ```
   enable-me
   ```

   Result: You can now perform system changes.
5. To run data enlarge, enter this command:

   ```
   data_enlarge
   ```

   Result: The virtual machine can now detect the newly allocated space.

Add a secondary disk to a virtual machine

If the main disk is not large enough when it is first imported, you can do this procedure to add a larger virtual data disk to the virtual machine (VM).

Before you begin

You should be familiar with managing virtual disks in your hypervisor environment. If you are not, refer to your hypervisor manual or contact your hypervisor's support service.

About this task

Note:
When you add a disk, use a disk type of SCSI or SATA that uses a storage type of:

- LSI Logic SAS
- LSI Logic, or
- SATA

IDE disks are not recommended.

Procedure

1. Add a disk to the *VM*.
2. Restart the *VM*.
3. In the *VM* console, to obtain the name of the disk devices, enter this command:

   ```
   sysctl kern.disks
   ```

4. Assuming **ada1** is the device disk added as a secondary disk (note that ada0 is the OS device), execute this command to move the data partition to it:

   ```
   data_move ada1
   ```

Add a monitoring interface to the virtual machine

By default, the virtual machine (VM) has one management network interface and one monitoring interface. Depending on deployment needs, it might be useful to add more monitoring interfaces to the sensor.

Procedure

1. If the *VM* is powered on, shut it down.
2. From the hypervisor configuration, add one or more network interfaces.
3. Add one or more monitoring interfaces in accordance with to the intended usage.

   The number of monitoring interfaces will depend on the intended usage.
4. Power on the *VM*.

Results

The new interface(s) is (are) now automatically recognized and used.



Chapter 4. Container installation



Docker installation

A Docker installation is a deployment of a sensor inside of an existing network component that supports Docker containers. If you want to take advantage of Nozomi Networks sensors can be installed in a Docker container.

Docker installations are best-suited for organizations that:

- Use robust network infrastructure components that support Docker containers
- Manage highly-distributed networks
- Aim to optimize acquisition, deployment, and management costs

A Docker container lets you install *N2OS* on embedded platforms that have a container engine on-board, such as:

- Switches
- Routers
- Firewalls
- *programmable logic controller (PLC)*s

It is also a good platform for tightly integrated scenarios where several products interact on the same hardware platform to provide a unified experience.

For all the other use cases, Nozomi Networks recommends one of these options:

- A physical installation
- A virtual installation

Commands

A Docker container installation has the same features as those that physical and virtual installation provide. However, a key difference is that you must use Docker commands to perform container provisioning **system** settings. Therefore, you cannot edit them from inside the container itself.

For example, the **hostname** must be set when you launch a new instance of the image.

Also, you must use volumes for the `/data` partition to make sure that the data will survive image updates.

The `network=host` Docker parameter permits the container to monitor the physical *network interface controller (NIC)*s on the host machine. However, by default it also permits the container to monitor all of the available interfaces. To restrict to a subset, create a `cfg/n2osids_if` file in the `/data` volume with the list of interfaces to monitor and use a comma to separate them. For example, `eth1,eth2`.

To stop the container, you can enter the command:

```
docker stop nozomi-rc
```

To execute the container, you can enter the command:

```
docker start nozomi-rc
```

Docker prerequisites

These are the prerequisites for a Docker installation.

In order to deploy Remote Collector on a Docker device, it must:

- Be based on AMD64 (or AArch64 architectures on ARMv8 in the case of the Cortex-A53)
- Available resources that are strictly to be devoted to the containers must match the table of recommended resource allocation

Docker container supportability for Remote Collector deployments:

- Docker version 24.x
- Docker buildX
- Overlay2 (best performance) or AUFS as storage backend
- TMPFS mount type
- Cisco IOS Cisco on C9300-48T 17.06.03 or Cisco IOx Local Manager 2.5.0.0
- Siemens LPE firmware version 1.00.00

Build a Docker image

Build a Docker image for your sensor.

Before you begin

Make sure that:

- Docker is installed. (Nozomi Networks has tested Docker versions 18.09 and 20.10.)
- BuildKit is installed. (Docker 18.09 or higher is required. To activate BuildKit, refer to the Docker BuildKit documentation.)

About this task

You can use variables to customize the container version build. Use the `--build-arg` command line switch to pass the variables to the Docker build command.

Procedure

To build the image, from the directory containing the artifacts, enter this command:

```
docker build -t nozomi-rc .
```

An example of a customization:

```
docker build --build-arg APT_PROXY=10.0.0.17:5843 -t rc
```

To customize the method the sensor uses to communicate, these parameters can be given in the build command:

Parameter	Default value	Description
APT_PROXY	None	Proxy to be used to download container packages
N2OS_HTTP_PORT	80	Specify custom *hypertext transfer protocol (HTTP)* web port
N2OS_HTTPS_PORT	443	Specify custom *HTTPS* web port

Result: The image is created.

Install a sensor in a Docker container

Docker is the approved software for a container installation on Nozomi Networks Operating System (N2OS). Do this procedure to run a Nozomi Networks sensor in a Docker environment.

Procedure

To run the image, enter a command such as:

```
docker run --hostname=nozomi-rc --name=nozomi-rc \
--volume=<path_to_data_folder>:/data --network=host \
-d nozomi-rc
```

Note:
`<path_to_data_folder>` is the path to a volume where the sensor's data will be stored, and saved for future runs.

The sensor will monitor all network interfaces available to the container. Use the --network=host option to give the container full access to all network interfaces of the host computer.

Update a Docker container

You can update the Docker container to a newer release of the Nozomi Networks Operating System (N2OS).

Procedure

1. Build a new version of the *N2OS* image.
2. Stop the current running containers.
3. Destroy the current running containers
4. Start a new container with the updated image.

Results

Data is automatically migrated to the new version.

Remote Collector container installation on the Cisco Catalyst 9300

It is possible to install the Remote Collector container on the Catalyst 9300 switch.

Before you install a Remote Collector container on the Cisco Catalyst 9300, you need to have:

- Extensive knowledge of IOS, IOx, and the ioxclient program
- Knowledge of the Cisco ioxclient (see the official Cisco documentation)
- Docker knowledge (see the official Docker documentation)
- *SSH* access to the Catalyst 9300
- Privileged access to the Catalyst 9300 with `enable` password
- Remote Collector container version must be present on your registry, or on the local Docker cache

IOS and IOx supported versions:

- Cisco IOS Cisco on C9300-48T 17.06.03
- Cisco IOx Local Manager 2.5.0.0

Minimum operational requirements:

- Catalyst 9300 must be enabled to host a container, a second storage can be required
- ioxclient program

Important:
The supported configuration is provided for the exclusive use of the Catalyst 9300 container subsystem by the Remote Collector container. No other containers can run at the same time. This configuration will use all of the *CPU* and *RAM* available for the container's subsystem.

Important:
All of the commands and configurations proposed in this documentation regarding the Catalyst 9300 are only examples. A qualified network administrator should verify all of the commands, which you can modify to suit the configuration that you are running. Incorrect configurations and commands on the Catalyst 9300 can make it unusable and can cause network disruptions.

Important:
You should change the `192.0.2.0/24` network, as it is for documentation purposes only.

Table 2. Used parameters

Parameter	Used value	Description
appid	NozomiNetworks_RC	The Remote Collector container name
guest-ipaddress	192.0.2.10	The Remote Collector container IP address
app-defaultgateway	192.0.2.1	The default gateway for the provided network
$VERSION	n.a.	To be filled with the Remote Collector version

Prepare the Remote Collector container as a IOx packet

This procedure describes the preparation required to set up the Remote Collector container on the Catalyst 9300.

Procedure

1. Create a `package.yaml` file with the contents that follow:

```
descriptor-schema-version: "2.10"
info:
    name: NozomiNetworks_RC
    version: latest
app:
    cpuarch: x86_64
    resources:
        persistent_data_target: "/data"
        network:
        - interface-name: eth0
        - interface-name: eth1
          mirroring: true
        profile: custom
        cpu: 7400
        memory: 2048
        disk: 4000
    startup:
        rootfs: rootfs.tar
        target: ["/usr/local/sbin/startup-container.sh"]
        user: admin
        workdir: /data
    type: docker
```



Note:
ioxclient uses the above configuration to build the Remote Collector container for IOx. It enables mirrored ports on the Cat9300 IOx backplane on to the container's eth1 port and set `/data` as persistent storage on the Catalyst 9300. Other input ports are not needed for the Remote Collector.

2. To build the Remote Collector container for the IOx package in the same directory as that of the `package.yaml`, enter the command:

```
ioxclient docker package --skip-envelope-pkg your-containerregistry.
com/NozomiNetworks_RC:"$VERSION" .
```

> **Note:**
> This creates the `package.tar`. You must upload this file directly onto the IOx as covered in the Cisco IOx documentation.

> **Important:**
> After every Catalyst 9300 configuration change or redeploy, you must stop the app and activate/start it again.

3. Import the previously generated package in the Catalyst 9300 IOx subsystem as described in the Cisco IOx documentation.
4. To activate and start the application, on the Catalyst 9300 *SSH* console, enter these commands:

```
app-hosting stop appid NozomiNetworks_RC
app-hosting activate appid NozomiNetworks_RC
app-hosting start appid NozomiNetworks_RC
```

5. To access the container through the Catalyst 9300 console, enter the command:

```
app-hosting connect appid NozomiNetworks_RC session
```

6. Configure the Remote Collector (on page 73).

Configure the Remote Collector container on the Cisco Catalyst 9300

This procedure describes how to set up the Remote Collector container on the Cisco Catalyst 9300.

Procedure

1. To go to privileged mode in the Catalyst 9300, enter the command:

```
enable
```

2. To set up the Catalyst 9300 AppGigabitEthernet interface in trunk mode, enter the command:

```
conf t

interface AppGigabitEthernet 1/0/1
switchport mode trunk
exit
```

3. To configure the IOx to host the container, enter the command:

```
app-hosting appid NozomiNetworks_RC
    app-vnic management guest-interface 0
    guest-ipaddress 192.0.2.10 netmask 255.255.255.0
    app-default-gateway 192.0.2.1 guest-interface 0
    app-vnic AppGigabitEthernet trunk
app-hosting appid NozomiNetworks_RC
    app-vnic AppGigabitEthernet trunk
    guest-interface 1
    mirroring
end
```

Results

The Remote Collector container is now set up on the Cisco Catalyst 9300.



Chapter 5. Basic configuration



Do the basic configuration

Before you can use the Nozomi Networks Operating System (N2OS) software, you must do the basic configuration.

Before you begin

You have installed the *N2OS* software.

About this task

After you have done this procedure, your system management interface will be set up and reachable:

- As a text console through *SSH*

Note:
The configuration of more than one **mgmt** interface is not supported.

Note:
You can use either a serial console (for physical sensors) or the text hypervisor console (for virtual sensors). This is dependent on the sensor model.

Note:
The sensor's shell has two users: **admin** and **root**. You should use **admin** to log in to both. If you want to elevate from **admin** to **root**, you should use the **admin** password. The **root** account does not have a separate password.

Procedure

1. The console shows a prompt with the text `N2OS - login:`. Type **admin** and then press **Enter**.

   > **Note:**
   > In a virtual sensor, you are instantly logged in, as no password is set by default. In physical sensors, **nozominetworks** is the default password.

   > **Note:**
   > If you ever need to change the password, you can use the **change_password** command.

2. To elevate your privileges, enter the command:

   ```
   enable-me
   ```

   Result: Your privileges have been elevated.

3. To launch the initial configuration wizard, enter the command:

   ```
   setup
   ```

4. For virtual sensors, at the prompt, choose the **admin** password first.

   > **Attention:**
   > You should select a strong password as this will permit the **admin** user to access the sensor through *SSH*.

5. In the menu, select **Network Interfaces**.
6. Select the management interface, then select **Enter**.
7. Choose a method to configure the *internet protocol (IP)* address.

   Choose from:

   - To configure all automatically, move the cursor to **DHCP** and select **Enable**, then (go to step 9 (on page 48))
   - Do a static configuration (go to step 8 (on page 48))

8. Do a static configuration.
   a. Move the cursor to **ipaddr** and edit the values for the *IP* address.
   b. Move the cursor to **netmask** and edit the values for the netmask.
9. Move the cursor to **X Save/Exit** and select **Enter**.
10. In the menu, select **Default Router/Gateway**.

11. Enter the *IP* address of the default gateway.

> **Important:**
> A valid default gateway **must** be configured during setup to ensure correct system operation. Failure to do so can result in malfunction, erratic behavior, loss of connectivity, or sensor isolation. If your network uses static addressing, make sure that this value reflects the correct route for outbound traffic.

12. Press **tab** and then select **Enter** to save and exit.
13. In the menu, select **DNS nameservers**.
14. Configure the *IP* addresses of the *domain name server (DNS)* servers.
15. Move the cursor to **X Exit** and select **Enter**.

    Result: The basic configuration is complete.



Chapter 6. Additional settings



Network requirements

Operator access to Guardian, CMC, and RC

The required ports and protocols for operator access to Guardian, CMC, and RC.

Table 3. Operator access to Guardian, CMC, and RC

Port	Protocol	Source	Destination	Purpose	Direction
tcp/443	https	Operator	Guardian/CMC	Operator's https access to Guardian/ CMC Web UI	Inbound
tcp/22	ssh	Operator	Guardian/ CMC/RC	Operator's ssh access to Guardian/ CMC/RC shell	Inbound
UDP/48888	Multiple protocols	Guardian and monitored devices	Broadcast	Discovery	Bi-directional
UDP/1911	Fox	Guardian and monitored devices	Broadcast	Discovery	Bi-directional

Communications between Guardian, CMC, and RC

The required ports and protocols for communications between Guardian, CMC, and RC.

Table 4. Communications between Guardian, CMC, and RC

Port	Protocol	Source	Destination	Purpose
tcp/443	https	Guardian/ CMC	CMC	Sync from Guardian to CMC or between CMCs of different tiers in the hierarchy
tcp/443	https	RC	Guardian	Sync from Remote Collector to Guardian

Table 4. Communications between Guardian, CMC, and RC (continued)

Port	Protocol	Source	Destination	Purpose
tcp/6000	proprietary (over TLS)	RC	Guardian	Transmission of monitored traffic from Remote Collector to Guardian

Operator access to Vantage

The required ports and protocols for operator access to Vantage.

Table 5. Operator access to Vantage

Port	Protocol	Source	Destination	Purpose
tcp/443	https	Operator	Vantage	Operator's https access to Vantage Web UI

Communications between Guardian, CMC, and Vantage

The required ports and protocols for communications between Guardian, CMC, and Vantage.

Table 6. Communications between Guardian, CMC, and Vantage

Port	Protocol	Source	Destination	Purpose
tcp/443	https	Guardian/ CMC	Vantage	Sync from Guardian or CMC to Vantage

Install a CA certificate

If an issuing certificate authority (CA) for the hypertext transfer protocol secure (HTTPS) certificate is not immediately trusted, you will need to install a certificate authority (CA) certificate to the Nozomi Networks Operating System (N2OS) sensor.

Before you begin

Make sure that:

- Your intermediate *certificate authority (CA)* and Root *CA* certificates are combined
- The certificate must be in *privacy-enhanced mail (PEM)* format

Note:
These formats are not supported:

- Distinguished Encoding Rules (DER)
- PKCS#12

Procedure

1. Upload the *CA* certificate to the sensor.
   a. Change the name of the *CA* certificate to `cert.crt`
   b. Open a terminal.
   c. To upload, enter this command:

   ```
   scp cert.crt admin@<sensor_ip>:/data/tmp
   ```

2. Log into the console, either directly or through *SSH*.
3. To go to privileged mode, enter this command:

   ```
   enable-me
   ```

   Result: You can now perform system changes.

4. Change directory into the `/data/tmp` folder.
5. To add the *CA* certificate to the trust store, enter this command:

   ```
   n2os-addcacert cert.crt
   ```

Results

The sensor now trusts the imported *CA* certificate. You can now use it to secure *HTTPS* communication to the sensor.

Configure certificate rotation

When sensors are not provisioned with custom Remote Collector certificates, the automatically generated certificates are periodically rotated to ensure that communication between sensors does not break due to expired certificates.

About this task

The default rotation period is ninety days. You can configure a different value, with a minimum of one day and a maximum of one year.

If you provision custom certificates, automatic rotation does not apply. For more details, see Configure custom certificates (on page 85).

Procedure

To set the rotation interval, enter this command:

```
conf.user rc_certificate_rotation_task rotation_interval DAYS
```

Result: The rotation interval is updated. Certificates are automatically renewed at the new interval.

Enable SNMP

To monitor the health of the Nozomi Networks Operating System (N2OS) sensor, you need to enable the simple network management protocol (SNMP) daemon.

About this task

The current *simple network management protocol (SNMP)* daemon supports versions v1, v2c and v3. This feature is not available with a container installation.

Procedure

1. Log into the console, either directly or through *SSH*.
2. To go to privileged mode, enter this command:

   ```
   enable-me
   ```

   Result: You can now perform system changes.

3. Edit these variables as necessary:
   - location
   - contact
   - community

   Important:

   For community, it is important to use a strong password.

4. Change the value of other variables as necessary.



Note:
For SNMP v3 User-Based Security Model (USM), uncomment the following sections in `/etc/snmpd.conf` to create a user bsnmp and set privacy and encryption options to SHA message digests and AES encryption for this user:

```
engine := 0x80:0x10:0x08:0x10:0x80:0x25
snmpEngineID = $(engine)
```

```
user1 := "bsnmp"
user1passwd :=
 0x22:0x98:0x1a:0x6e:0x39:0x93:0x16: ... :0x05:0x16:0x33:0x38:
0x60
```

```
begemotSnmpdModulePath."usm" = "/usr/lib/snmp_usm.so"
```

```
%usm
```

```
usmUserStatus.$(engine).$(user1) = 5
usmUserAuthProtocol.$(engine).$(user1) =
 $(HMACSHAAuthProtocol)
usmUserAuthKeyChange.$(engine).$(user1) = $(user1passwd)
usmUserPrivProtocol.$(engine).$(user1) = $(AesCfb128Protocol)
usmUserPrivKeyChange.$(engine).$(user1) = $(user1passwd)
usmUserStatus.$(engine).$(user1) = 1
```

5. Edit the `/etc/rc.conf` file with this line:

```
bsnmpd_enable="YES"
```

6. To start the service, enter this command:

```
service bsnmpd start
```

7. If you enabled the User-Based Security Model (USM) in step 3 (on page 57), replace the default value for the `user1passwd` *variable*.
    a. To convert the SHA or MD5 output to exe format, enter this command:

    ```
    sh -c "SNMPUSER=bsnmp SNMPPASSWD=<newpassword>
     SNMPAUTH=<sha|md5> SNMPPRIV=<aes|des> bsnmpget -v 3 -D -K -o
     verbose"
    echo <SHA output> | sed 's/.\{2\}/:0x&/g;s/^.\{6\}//g'
    ```

    b. To restart the service, enter the command:

    ```
    service bsnmpd restart
    ```

8. To save all of the settings, enter this command:

    ```
    n2os-save
    ```

9. To check the functionality, run a test command from an external system (the `<sensor_ip>` has to be reachable). For example, for the USM case, with the default values in the `/etc/snmpd.conf` file, use a command similar to this:

    ```
    snmpstatus -v3 -u bsnmp -a SHA -A <password> -x AES -X <password> -l
     authPriv <sensor_ip>
    ```

Results

SNMP has been enabled.

Internal firewall configuration

It is possible to configure the settings of an internal firewall to restrict access to specific items.

You can limit access to the:

- Management interface
- *SSH* terminal
- *SNMP* service
- *internet control message protocol (ICMP)*

Note:
It is only possible to do this for physical and virtual installations. It is not possible for container installations.

Note:
To limit access to these services, you must use the *command-line interface (CLI)* to add the required configurations.

Note:
The default settings permit connections from any *IP* address. The system ignores lines with invalid *IP* addresses.

Important:
You should use caution when changing internal firewall rules. This is because you can lose access to the device administration interface. In the event of an error, console access is required to fix the rules.

The table below gives the configuration settings that let you fine-tune the firewall rules.

Table 7. Internal firewall configuration settings

Parameter	Description
system firewall icmp	Configure acl for icmp protocol
system firewall https	Configure acl for http and https services
system firewall ssh	Configure acl for ssh service
system firewall snmp	Configure acl for snmp service

Management-interface packet-rate protection

A description of internal packet-rate protection, which is enabled by default.

The management interface of the physical and virtual sensors have internal packet rate protection enabled by default.

Malicious hosts are banned for five minutes if they try to send more than 1024 packets within five seconds.

To show a list of blocked *IP* addresses, you can use the `n2os-firewall-show-block` command.

To unblock a single *IP* address, you can use the `n2os-firewall-unblock <ip_address_to_unblock>` command.

Disable internal packet-rate protection of the management interface

If necessary, do this procedure to disable internal packet-rate protection in the management interface.

Procedure

1. In the top navigation bar, select 

   Result: The administration page opens.

2. In the **Settings** section, select **CLI**.

   Result: The **CLI** page opens.

3. In the *CLI*, enter this command:

   ```
   conf.user configure system firewall disable_packet_rate_protection
    true
   ```

4. Log into the text console, either directly, or through *SSH*.
5. To apply the new settings, enter this command:

   ```
   n2os-firewall-update
   ```

Results

The internal packet-rate protection has been disabled.

Configure IPv6

You can configure IPv6 to access physical and virtual sensors. This is not possible for container installations.

Procedure

1. Log into the text console, either directly, or through *SSH*.
2. To enable IPv6 on the management interface, enter this command:

   ```
   n2os-setupipv6
   ```

3. To reboot the sensor, enter this command:

   ```
   reboot
   ```

4. Wait for the sensor to reboot.
5. To retrieve the address, use a command such as:

   ```
   ifconfig
   ```

   Note:
   You can now enclose the address in square brackets [ ] to access the sensor *user interface (UI)*. Similarly, you can configure sensors to synchronize towards a *CMC* or another sensor in *high availability (HA)* mode. To do this you can specify the IPv6 address in square brackets [ ].

Results

IPv6 has been configured.

Enable 802.1x on the management interface

This topic describes how to enable 802.1x support for the management interface. Configuration of the RADIUS server and the creation of possible certificates are not discussed

Before you begin

- Make sure that you have serial access to the sensor as part of this configuration is performed via serial console
- If the 802.1x is already configured, switch side and ports are already closed. Make sure that you have a network patch to reach the sensor through a direct network connection
- If the authentication process is through *TLS* certificates, confirm that you have `ca.pem`, `client.pem`, and `client.key` files, as well as the `client.key` unlock password
- If the authentication process is through the *protected extensible authentication protocol (PEAP)*, confirm that you have the identity and password

Procedure

1. Log into the console, either directly or through *SSH*.
2. To go to privileged mode, enter this command:

   ```
   enable-me
   ```

   Result: You can now perform system changes.

3. To create the directory `/etc/wpa_supplicant_certs` and change the directory permissions to `755`, enter this command:

   ```
   mkdir /etc/wpa_supplicant_certs
   chmod 755 /etc/wpa_supplicant_certs
   ```

   Important:
   You must use this exact directory name. No other name is permitted.

4. To create the file `/etc/wpa_supplicant.conf`, enter this command:

   ```
   vi /etc/wpa_supplicant.conf
   ```

   Important:
   You must use this exact file name. No other name is permitted. If necessary, you should rename the file to this name.

5. Examples of `wpa_supplicant.conf` are shown below:
   - Configuration for *PEAP* authentication:

```
ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=0
eapol_version=1
ap_scan=0
network={
    ssid="NOZOMI8021X"
    key_mgmt=IEEE8021X
    eap=PEAP
    identity="identity_for_this_guardian_here"
    password="somefancypassword_here"
}
```

   - Configuration for *TLS* authentication:

```
ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=0
eapol_version=1
ap_scan=0
network={
    ssid="NOZOMI8021X"
    key_mgmt=IEEE8021X
    eap=TLS
    identity="client"
    ca_cert="/etc/wpa_supplicant_certs/ca.pem"
    client_cert="/etc/wpa_supplicant_certs/client.pem"
    private_key="/etc/wpa_supplicant_certs/client.key"
    private_key_passwd="somefancypassword_private_key_here"
}
```

6. For *TLS* authentication, use Ethernet to connect to the sensor and copy the required files to the expected location.



Note:
If the sensor is not reachable via *SSH* using the actual network, we suggest that you configure the `mgmt` interface with a temporary *IP* address and connect the sensor with a direct Ethernet patch cable.

7. **Note:**
   If you are using *PEAP* authentication, you can skip the next step.

   For *TLS* authentication, upload the certificate files to the sensor with an *SSH* client in the `/etc/wpa_supplicant_certs/` folder.

   ```
   scp ca.pem client.pem client.key admin@<sensor_ip>:/tmp/
   ```

8. In the sensor serial console, with elevated privileges, move the files to the expected location:

   ```
   mv /tmp/ca.pem /tmp/client.pem /tmp/client.key /etc/
   wpa_supplicant_certs
   ```

9. **Note:**
   If you are using *PEAP* authentication, you can skip the next step.

   In the sensor serial console, with elevated privileges, to change the certificate permission to 440, enter these commands:

   ```
   cd /etc/wpa_supplicant_certs
   chown root:wheel ca.pem client.pem client.key
   chmod 440 ca.pem client.pem client.key
   ```

10. In the sensor serial console, with elevated privileges, to change the `/etc/rc.conf` file, enter the details that follow:

    ```
    wpa_supplicant_flags="-s -Dwired"
    wpa_supplicant_program="/usr/local/sbin/wpa_supplicant"
    ```

11. To change the `ifconfig_mgmt` entry in the `/etc/rc.conf` file, add the prefix WPA.

    Note:
    If the sensor was configured with a direct Ethernet patch cable, you can now configure the production-ready *IP* address and connect the sensor to the switch. For example, if the sensor *IP* address is `192.168.10.10`, the entry will be similar to:

    ```
    ifconfig_mgmt="WPA inet 192.168.10.10 netmask 255.255.255.0"
    ```

12. To save all of the settings, enter this command:

```
n2os-save
```

13. To reboot the system, enter this command:

```
shutdown -r now
```

14. Wait for the system to reboot.
15. Log in to the sensor.

16. Enter the command: `ps aux |grep wpa` You should receive output similar to the following:

```
root 91591 0.0 0.0 26744 6960 - Ss 09:59
0:00.01 /usr/local/sbin/wpa_supplicant -s -Dwired -B -i mgmt
 -c /etc/
wpa_supplicant.conf -D wired -P /var/run/wpa_supplicant/mgmt.pid
```

17. You can check the status of the `wpa_supplicant` with the `wpa_cli -i mgmt status` command. For example:

```
root@guardian:~# wpa_cli -i mgmt status
bssid=01:01:c1:02:02:02
freq=0
ssid=NOZOMI8021X
id=0
mode=station
pairwise_cipher=NONE
group_cipher=NONE
key_mgmt=IEEE 802.1X (no WPA)
wpa_state=COMPLETED
ip_address=192.168.1.2
address=FF:FF:FF:FF:FF:FF
Supplicant PAE state=AUTHENTICATED
suppPortStatus=Authorized
EAP state=SUCCESS
selectedMethod=13 (EAP-TLS)
eap_tls_version=TLSv1.2
EAP TLS cipher=ECDHE-RSA-AES256-GCM-SHA384
tls_session_reused=0
eap_session_id=0dd52aaeaa2aa3aa4deaac6aaafc65edbfa58cdffecff6ff4[.
..]
uuid=8a31bd80-1111-22aa-ffff-abafa0a9afa6
```

USB port disablement

The Nozomi Networks Operating System (N2OS) does not support universal service bus (USB) ports.

We do not recommend that you connect external disks or other devices to the *universal serial bus (USB)* port of a sensor, as we do not guarantee that they will work.

A *USB* keyboard will work in a hardened configuration. However, the **Ctrl+Alt+Del** shortcut will not work.

You can use readily available, external tools to physically block a *USB* port.

Chapter 7. Deployment



Connect a Remote Collector to Guardian

Once you have configured a remote Collector, you can connect it to a Guardian. You configure a Remote Collector through a terminal secure shell (SSH) or console.

Before you begin

Make sure that you have completed the basic configuration (on page 47).

Procedure

1. Log into the Guardian's console, either directly or through *SSH*.
2. Enter the command:

   ```
   n2os-enable-rc
   ```

   Note:
   This open port 6000 on the firewall, which allows the Remote Collector to send the traffic it sniffs. A new interface called `remote-collector` shows in the Guardian's list of `Network Interfaces`.

   Result: Port 6000 on the firewall opens.

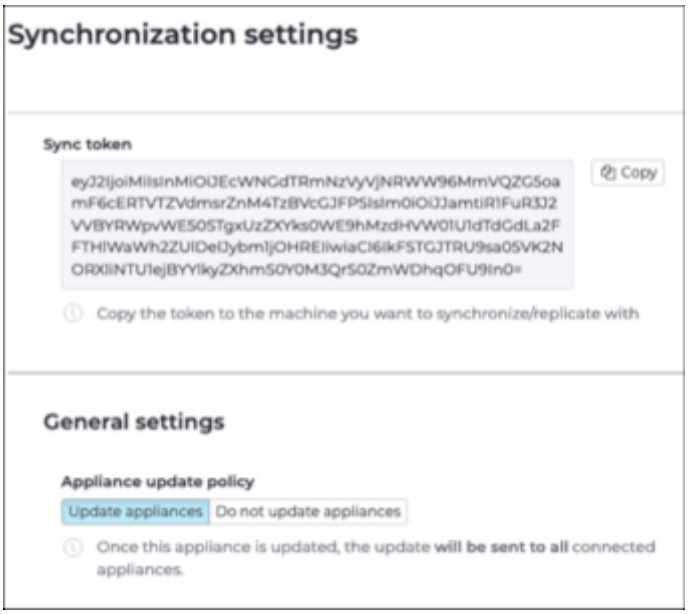
3. In the top navigation bar, select ⚙

   Result: The administration page opens.

4. In the **Settings** section, select **Synchronization settings**.

   Result: The **Synchronization settings** page opens.

5. To the right of the **Sync token** section, select **Copy**.



Result: Guardian is ready to connect to a Remote Collector.

6. Configure the Remote Collector (on page 73).

Configure the Remote Collector

To configure a Remote Collector, use the terminal (secure shell (SSH) or console).

About this task

The Remote Collector has a *text-based user interface (TUI)* to help you with this phase.

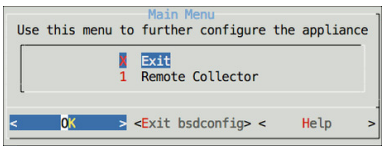
Procedure

1. Enter the command:

   ```
   n2os-tui
   ```

   Result: The *TUI* opens.

2. In the menu, select **Remote Collector**.

   Result: The **Remote Collector Management** menu opens.

3. In the menu, select **Set Guardian Endpoint**.

4. Enter the *IP* address of the Guardian that you wish to connect to.
5. In the menu, select **Set Connection Sync Token**.
6. In the **Sync Token** field, paste the sync token that you previously copied from Guardian.
7. **Optional:** In the menu, select **Set BPF Filter**.
8. In the **BPF Filter** field, enter the necessary value.
9. Exit from the *TUI*.

Set the time zone

To set the time zone of the sensor, edit the `/data/cfg/n2os.conf.user` file.

Procedure

1. Log into the console, either directly or through *SSH*.
2. Use `vi` or `nano` to edit the `/data/cfg/n2os.conf.user` file.
3. Use the *internet assigned numbers authority (IANA)* format to add a line in the file that states the applicable time zone.

   Note:
   For example: `system time tz Australia/Brisbane`

4. Reboot the sensor.

Results

The time zone has been configured.

Enable bandwidth throttling

You can specify the maximum amount of allowed traffic to limit the bandwidth that a sensor's management port has at its disposal (for access and updates).

Before you begin

Make sure that you have:

- Enabled the Remote Collector
- The correct privileges

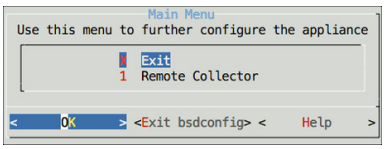
This step assumes that you have the correct privileges and the remote collector is enabled.

Procedure

1. Enter the command:

   ```
   n2os-tui
   ```

2. In the menu, select **Remote Collector**.

   Result: The **Remote Collector Management** menu opens.

3. In the menu, select **Set traffic shaping bandwidth**.

4. Enter a value for the maximum bandwidth to use.

   Note:

   For example, 2 *megabyte (MB)* will set a maximum of 2 *Megabit per second (Mb/s)*.

5. **Optional:** If necessary, exclude specific *IP* addresses from the bandwidth limitation.
    a. In the menu, select **Set traffic shaping exclusions**.
    b. Enter the *IP* address that you want to exclude.
6. In the menu, select **Set max bandwidth kb/s**.

> **Note:**
> For Remote Collectors, you can specify the maximum amount of allowed bandwidth to limit the bandwidth for the traffic sniffed and forwarded to the Guardian. You can do this without impacting other connections on the management port,

7. Enter the maximum bandwidth in *kilobit per second (kb/s)*.
8. Exit the *TUI*.

Results

Bandwidth throttling has been enabled.

Enable multiplexing

In addition to a primary Guardian, Remote Collectors can multiplex traffic to a set of secondary Guardians. Each Guardian receives the same traffic information from the Remote Collector.

About this task

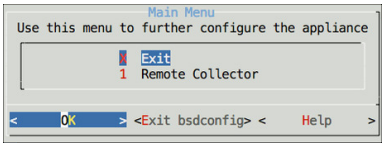
To enable multiplexing, configure a secondary Guardian. Although every Guardian receives the same traffic information, only the primary Guardian is authorized to change its settings. At each sync, in the event of a communication failure with the primary Guardian, the secondary Guardian acquires the configuration capabilities.

Procedure

1. Enter the command:

   ```
   n2os-tui
   ```

2. In the menu, select **Remote Collector**.

   Result: The **Remote Collector Management** menu opens.

3. In the menu, select **Set secondary endpoint**.

4. Enter the *IP* address and token of the secondary endpoint.
5. In the menu, select **Set secondary token**.
6. Enter the sync token for the secondary Guardian.
7. Exit from the *TUI*.

Results

Multiplexing has been enabled.

Configure the site name and description

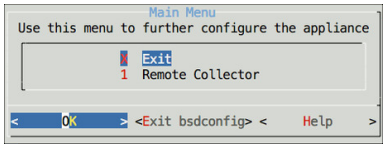
You can set a site name and description for a Remote Collector to help identify it.

Procedure

1. Enter the command:

   ```
   n2os-tui
   ```

2. In the menu, select **Remote Collector**.

   Result: The **Remote Collector Management** menu opens.

3. In the menu, select **Set site**.

4. Enter the **Site name**.
5. **Optional:** In the menu, select **Set description**.
6. **Optional:** Enter a description.
7. Exit from the *TUI*.

Results

The site name and description have been configured.

Set the compression strategy from a shell console

You can set a compression strategy to compress the traffic that the Remote Collector generates.

About this task

Possible values are:

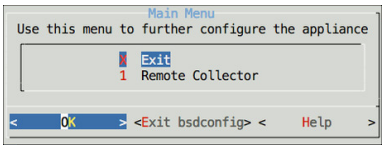
- 'raw': (default) sends traffic without compression. This strategy saves *CPU* time if the traffic is hard to compress
- 'zstd' sends traffic compressed with the zstd algorithm
- 'zraw' sends raw traffic compressed with the zstd algorithm, thus achieving greater efficiency

Procedure

1. Enter the command:

   ```
   n2os-tui
   ```

2. In the menu, select **Remote Collector**.

   Result: The **Remote Collector Management** menu opens.

3. In the menu, select **Set strategy**.

4. In the **Set strategy** field, enter a strategy value.
5. Exit from the *TUI*.

Results

The compression strategy has been set.

Set the compression strategy from Guardian or CMC

Setting a compression strategy lets you compress traffic that the Remote Collector generates.

About this task

Possible values are:

- 'raw': (default) sends traffic without compression. This strategy saves *CPU* time if the traffic is hard to compress
- 'zstd' sends traffic compressed with the zstd algorithm
- 'zraw' sends raw traffic compressed with the zstd algorithm, thus achieving greater efficiency

Procedure

1. In the top navigation bar, select **Sensors**.

   Result: The **Sensors** page opens.

2. In the list, find and select the applicable Remote Collector.



   Result: The details for the Remote Collector show in the information pane on the right.

3. In the **Remote sensor strategy**, the current strategy shows.



4. To the right of the **Remote sensor strategy** line, select the icon.

   Result: A dialog shows.

5. From the dropdown, select the strategy that you want to use.



6. Select **Ok**.

> **Note:**
> It might take a few minutes to complete the exchange.

Results

The compression strategy has been set.

Enable traffic forwarding

The previous steps enable the Remote Collector to communicate with the Guardian sensor, but traffic forwarding requires the two to exchange the certificates required for encrypting the sniffed traffic being forwarded.

About this task

This procedure explains the simplest way to configure the certificate exchange. For an alternative approach, see Configure custom certificates (on page 85).

Procedure

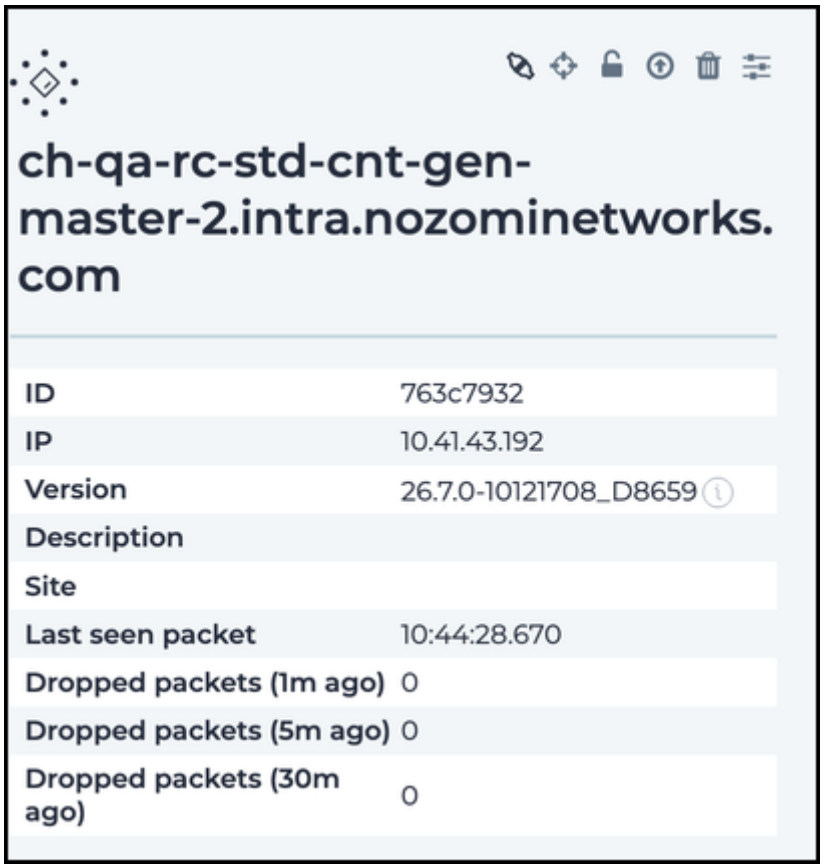
1. In the top navigation bar, select **Sensors**.

   Result: The **Sensors** page opens.

2. Select the new Remote Collector.

   Result: A pane opens on the right.

3. View the pane on the right.



Note:
The **Last seen packet** property shows whether traffic is being forwarded or not.

4. At the top of the pane, select the icon to allow the sensor.

   Result: A dialog shows. To confirm, select **Yes**.

5. To the right of the **Not connected** label, select the ⇄ icon.
6. At the top of the pane, select the **Live** toggle  to on.

> **Note:**
> It takes a few minutes to complete the exchange and the last step is completed only after the Remote Collector sends the first encrypted packet to the Guardian sensor. If no traffic is being sniffed, and therefore forwarded, the procedure remains stuck in the connecting step with a spinning wheel.

Result: The button changes into a spinning wheel. After few minutes, once the procedure is complete, the date and time of the last seen packet shows.

Configure custom certificates

You can provision custom Remote Collector certificates on both the Guardian and the Remote Collector sensors to replace the automatically generated self-signed certificates.

Before you begin

Make sure that you have:

- Both the certificate and the key file in *PEM* format
- A complete certificate chain

> **Important:**
> This feature is not supported in containerized environments.

About this task

To add a private *CA* to the system's trust store, see Install a CA certificate (on page 55).

You must perform this procedure on both the Guardian and the Remote Collector using their respective certificates.

Procedure

1. Select the new Remote Collector.

   Result: A pane opens on the right.

2. At the top of the pane, select the icon to allow the sensor.

   Result: A dialog shows. To confirm, select **Yes**.

3. Change the name of the certificate file to `rc_nozomi.crt`
4. Change the name of the key file to `rc_nozomi.key`
5. Upload the certificate and key files to the sensor `/data/tmp` folder
6. To enable the certificate, enter this command:

   ```
   n2os-addtlscert --rc rc_nozomi.crt rc_nozomi.key
   ```

7. Put the Guardian full **certificate chain** (that is, `/data/ssl/rc_nozomi.crt` and the *CA* certificate) under `/data/ssl/trusted_nozomi.crt` on the Remote Collector.

   Result: The Remote Collector certificate is now accepted on the Guardian.

8. Put the Remote Collector full **certificate chain** (that is, `/data/ssl/rc_nozomi.crt` and the *CA* certificate) under `/data/ssl/trusted_nozomi.crt` on the Guardian.

   Result: The Guardian certificate is now accepted on the Remote Collector.

Disable a Remote Collector

You can disable Remote Collectors that are not in-use to make your environment more secure.

Procedure

1. Log into the Guardian Web *UI* that receives data from the Remote Collector.
2. In the top navigation bar, select **Sensors**.

   Result: The **Sensors** page opens.

3. Select the applicable sensor in the table and select the  icon.

   Result: The Remote Collector has been deleted from the environment.

4. **Optional:** Remove all of the Remote Collectors from the environment.
   a. Open a shell console on the Guardian that is receiving data from the Remote Collector.
   b. To go to privileged mode, enter this command:

   ```
   enable-me
   ```

   a. Enter the command:

   ```
   n2os-disable-rc
   ```

   Result: All the Remote Collectors in your environment have been disabled.

Results

Your environment is now more secure.

Chapter 8. Compatibility Reference



Configuring SSH profiles

Different secure shell (SSH) cipher configurations are necessary for system security and compatibility. N2OS provides various profiles with customized cipher parameters to meet specific requirements.

About this task

The **standard** profile offers the highest level of security and has been the system default since version 23.3. The **legacy** profile ensures compatibility with older systems and was the default configuration prior to version 23.3. The **ccn** profile complies with the Spanish CCN standards, while the **fips** profile adheres to *Federal Information Processing Standards (FIPS)*-approved ciphers.

Choose the profile that best suits your needs and follow the provided commands to change it.

To change the *SSH* profile, use the following commands:

Procedure

1. Log in to the console, and enter privileged mode with the command:

   ```
   enable-me
   ```

2. Type the following command to change the *SSH* profile:

   ```
   sysrc n2osssh_profile=<standard|legacy|ccn|fips|pre256>
   ```

3. Save the configuration change:

   ```
   n2os-save
   ```

4. Reboot the appliance to apply the new configuration:

   ```
   shutdown -r now
   ```

   To disable *SSH* access via password authentication, you can add the postfix `-nopwd` to a profile. For example:

   ```
   sysrc
    n2osssh_profile=<standard-nopwd|legacy-nopwd|ccn-nopwd|fips-nopwd|p
   re256-nopwd>
   ```

   CAUTION:
   With this configuration, *SSH* access will not be possible unless the *SSH* key is configured in the Web *UI*.

   For more details about how to add *SSH* keys, see **Add an SSH key for an admin user** in the **Administrator Guide**.

SSH compatibility

Tables that give configuration details for each profile.

Standard SSH protocols profile (since 25.6.0)

Function	Algorithms
Key exchange	mlkem768x25519-sha256 (since 26.6.0) sntrup761x25519-sha512@openssh.com curve25519-sha256 curve25519-sha256@libssh.org diffie-hellman-group16-sha512 diffie-hellman-group18-sha512 ecdh-sha2-nistp256 ecdh-sha2-nistp384 ecdh-sha2-nistp521
Ciphers	chacha20-poly1305@openssh.com aes128-ctr aes192-ctr aes256-ctr aes128-gcm@openssh.com aes256-gcm@openssh.com
MACs	umac-128-etm@openssh.com hmac-sha2-256-etm@openssh.com hmac-sha2-512-etm@openssh.com
Host key algorithms	ssh-ed25519 ecdsa-sha2-nistp256

Previous SSH protocols profile (from 23.3.0 to 25.5.0)

Function	Algorithms
Key exchange	sntrup761x25519-sha512@openssh.com curve25519-sha256 curve25519-sha256@libssh.org diffie-hellman-group-exchange-sha256 diffie-hellman-group16-sha512 diffie-hellman-group18-sha512

Function	Algorithms
Ciphers	chacha20-poly1305@openssh.com aes128-ctr aes192-ctr aes256-ctr aes128-gcm@openssh.com aes256-gcm@openssh.com
MACs	umac-128-etm@openssh.com hmac-sha2-256-etm@openssh.com hmac-sha2-512-etm@openssh.com
Host key algorithms	rsa-sha2-512 rsa-sha2-256 ssh-ed25519

Legacy SSH protocols profile (prior to 23.3.0)

Function	Algorithms
Key exchange	curve25519-sha256@libssh.org diffie-hellman-group18-sha512 diffie-hellman-group14-sha256 diffie-hellman-group16-sha512 diffie-hellman-group-exchange-sha256
Ciphers	chacha20-poly1305@openssh.com aes256-gcm@openssh.com aes128-gcm@openssh.com aes256-ctr aes192-ctr aes128-ctr
MACs	hmac-sha2-256 hmac-sha2-512 hmac-sha2-512-etm@openssh.com hmac-sha2-256-etm@openssh.com umac-128-etm@openssh.com

Function	Algorithms
Host key algorithms	ecdsa-sha2-nistp384 ecdsa-sha2-nistp521 ssh-rsa ssh-ed25519

CCN SSH protocols profile

Function	Algorithms
Key exchange	ecdh-sha2-nistp256 ecdh-sha2-nistp384 ecdh-sha2-nistp521 diffie-hellman-group18-sha512 diffie-hellman-group16-sha512
Ciphers	aes256-gcm@openssh.com aes128-gcm@openssh.com aes256-ctr aes192-ctr aes128-ctr
MACs	hmac-sha2-256-etm@openssh.com hmac-sha2-512-etm@openssh.com hmac-sha2-256 hmac-sha2-512
Host key algorithms	ecdsa-sha2-nistp384 ecdsa-sha2-nistp521 ecdsa-sha2-nistp256

For more details, see Supported SSH protocols in FIPS mode (on page 94).

HTTPS compatibility

Lists the hypertext transfer protocol secure (HTTPS) and transport layer security (TLS) protocols supported by Guardian and Remote Collector for secure communication. Describes the available cipher suites for each TLS version. Use this information to verify compatibility and maintain secure data exchange between components.

Supported HTTPS protocols (since 21.9.0)

TLS version	Cipher Suite Name (IANA/RFC)
TLS 1.2	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
TLS 1.3	TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256

Supported RC/Guardian data channel protocols (since 21.9.0)

TLS version	Cipher Suite Name (IANA/RFC)
TLS 1.2	ECDHE-RSA-AES128-GCM-SHA256 ECDHE-RSA-AES256-GCM-SHA384 DHE-RSA-AES128-GCM-SHA256 DHE-RSA-AES256-GCM-SHA384

Supported SSH protocols in FIPS mode

A list of all the secure shell (SSH) protocols that are supported in Federal Information Processing Standards (FIPS) mode.

Table 8. Supported SSH protocols in FIPS mode

Function	Algorithms
Key exchange	ecdh-sha2-nistp256
	ecdh-sha2-nistp384
	ecdh-sha2-nistp521
	diffie-hellman-group-exchange-sha256
	diffie-hellman-group14-sha256
	diffie-hellman-group16-sha512
	diffie-hellman-group18-sha512
Ciphers	aes256-gcm@openssh.com
	aes256-ctr
	aes128-gcm@openssh.com
	aes128-ctr
MACs	hmac-sha2-256-etm@openssh.com
	hmac-sha2-512-etm@openssh.com
	hmac-sha2-256
	hmac-sha2-512
Host key algorithms	ecdsa-sha2-nistp256
	ecdsa-sha2-nistp384
	ecdsa-sha2-nistp521
	rsa-sha2-256
	rsa-sha2-512

Chapter 9. Federal Information Processing Standards



Federal Information Processing Standards

A description of the use of Federal Information Processing Standards in the Nozomi Networks software.

You can configure the *N2OS* software to use the FIPS-140-2 approved cryptography module. The *National Institute of Standards and Technology (NIST)* develops *FIPS* for non-military American government agencies, and government contractors, to use in computer systems.

The FIPS-140 series specifies requirements for cryptography modules within a security system to protect sensitive, but unclassified, data.

To enable *FIPS* mode, you must install a *FIPS*-enabled license. To obtain a license, refer to your Nozomi Networks representative.

Important:

To enable *FIPS* mode, you must be running version N2OS 22.2.1 or later.

Important:

You can only connect *FIPS* products to other *FIPS* products. The use of mixed environments is not allowed.

Important:

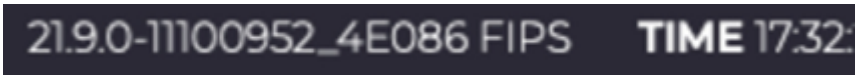
Enabling *FIPS* mode:

- If you are running a version of *N2OS* that is between 22.2.1 and 23.1.0, you will need a valid *FIPS* license for both Guardians and *CMC*s
- Beginning with version 23.1.0 or later, *FIPS* mode can be enabled on Guardians without a license, but packet sniffing will be disabled until a valid license is activated
- The order of enabling *FIPS* on either device does not affect functionality
- You need a *FIPS* license for *CMC*s and Remote Collectors. Upstream sensors will manage these licenses

FIPS mode status

When running in Federal Information Processing Standards (FIPS) mode, the sensor adds a FIPS indicator after the version string.

In the Web *UI*, look to see if `FIPS` is shown after the version string.



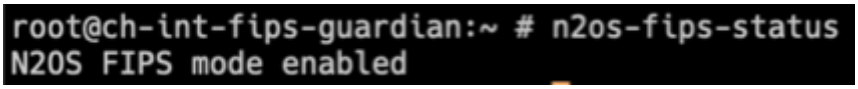
If you want to add FIPS to the version string, you can enter the command:

```
n2os-version
```



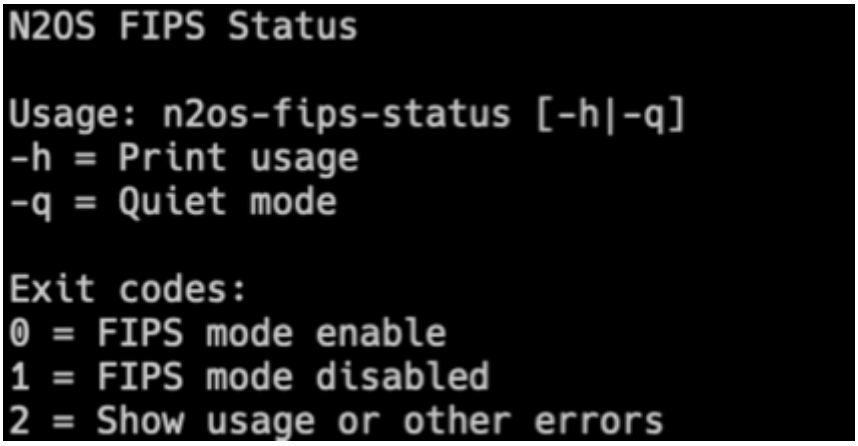
If you want to check the *FIPS* status, you can enter the command:

```
n2os-fips-status
```



To support additional parameters for extended usage, you can enter the command:

```
n2os-fips-status [-h | -q]
```



Enable FIPS mode

It is important that you follow this procedure to make sure that you enable Federal Information Processing Standards (FIPS) mode correctly.

About this task

When you switch to *FIPS* mode, local user Web *UI* passwords become invalid. To take advantage of *FIPS* encryption, you can use the `n2os-passwd` command to reset the passwords.

The `n2os-passwd <USER>` command takes several seconds to several minutes to prompt the user for a new password. On the R50 platform, the prompt may take up to three minutes.

Important:

When you enable *FIPS*, it is critical that you change the password for **EVERY** Web *UI* local user.

Procedure

1. Log into the console.
2. To go to privileged mode, enter this command:

   ```
   enable-me
   ```

   Result: You can now perform system changes.

3. **Note:**

   For *CMC*s, version 23.1.0 or later, it is not necessary to do the next step.

   Note:

   For Guardians, version 23.1.0 or later, you can also do the next two steps after you have enabled *FIPS*.

   To configure the *FIPS* license, enter the command:

   ```
   echo 'conf.user configure license fips xxxxxxxx' | cli
   ```

4. **Note:**
   For *CMC*s, version 23.1.0 or later, it is not necessary to do the next step.

   To restart the *intrusion detection system (IDS)*, enter the command:

   ```
   service n2osids stop
   ```

   Result: The *IDS* stops. After a few seconds, it will automatically start again.

5. To enable *FIPS* mode, enter the command:

   ```
   n2os-fips-enable
   ```

   Result: The system automatically reboots.

6. In the container edition, stop the current container and start a new one with the same settings.
7. To change the password for every user, enter the command:

   ```
   n2os-passwd <USER>
   ```

8. Log in to the sensor.
9. In the top navigation bar, select ⚙

   Result: The administration page opens.

10. In the **System** section, select **Updates and licenses**.

    Result: The **Updates and licenses** page opens.

11. In the **Current license** section for **FIPS**, make sure that the **License status** shows `ok`.
12. Do this procedure again for all *CMC*s and Guardians.

Disable FIPS mode

Do this procedure to disable Federal Information Processing Standards (FIPS) mode.

About this task

When you switch to *FIPS* mode, local user Web *UI* passwords become invalid. You can use the `n2os-passwd` command to reset the passwords.

Procedure

1. Log into the console, either directly or through *SSH*.
2. To go to privileged mode, enter this command:

   ```
   enable-me
   ```

   Result: You can now perform system changes.
3. To disable *FIPS* mode, enter the command:

   ```
   n2os-fips-disable
   ```
4. Reboot the system, or restart the container, to apply the changes.



Glossary



Amazon Machine Image

An AMI is a type of virtual appliance that is used to create a virtual machine for the Amazon Elastic Compute Cloud (EC2), and is the basic unit of deployment for services that use EC2 for delivery.

Amazon Web Services

AWS is a subsidiary of the Amazon company that provides on-demand cloud computing platforms governments, businesses, and individuals on a pay-as-you-go basis.

Asset Intelligence™

Asset Intelligence is a continuously expanding database of modeling asset behavior used by N2OS to enrich asset information, and improve overall visibility, asset management, and security, independent of monitored network data.

Berkeley Packet Filter

The BPF is a technology that is used in some computer operating systems for programs that need to analyze network traffic. A BPF provides a raw interface to data link layers, permitting raw link-layer packets to be sent and received.

Central Management Console

The Central Management Console (CMC) is a Nozomi Networks product that has been designed to support complex deployments that cannot be addressed with a single sensor. A central design principle behind the CMC is the unified experience, that lets you access information in a similar way as on the sensor.

Central Processing Unit

The main, or central, processor that executes instructions in a computer program.

Certificate Authority

A certificate, or certification authority (CA) is an organization that stores, signs, and issues digital certificates. In cryptography, a digital certificate certifies the ownership of a public key by the named subject of the certificate.

Command-line interface

A command-line processor uses a command-line interface (CLI) as text input commands. It lets you invoke executables and provide information for the actions that you want them to do. It also lets you set parameters for the environment.

Comma-separated Value

A CSV file is a text file that uses a comma to separate values.

Common Event Format

CEF is a text-based log file format that is used for event logging and information sharing between different security devices and software applications.

Common Vulnerabilities and Exposures

CVEs give a reference method information-security vulnerabilities and exposures that are known to the public. The United States' National Cybersecurity FFRDC maintains the system.

cURL

cURL is a command-line tool and library used to transfer data to or from a server. It supports many protocols, including HTTP, HTTPS, FTP, and more. In API contexts, curl is commonly used to test and interact with endpoints by sending requests and receiving responses.

Domain Name Server

The DNS is a distributed naming system for computers, services, and other resources on the Internet, or other types of Internet Protocol (IP) networks.

Federal Information Processing Standards

FIPS are publicly announced standards developed by the National Institute of Standards and Technology for use in computer systems by non-military American government agencies and government contractors.

Fully qualified domain name

An FQDN is a complete and specific domain name that specifies the exact location in the hierarchy of the Domain Name System (DNS). It includes all higher-level domains, typically consisting of a host name and domain name, and ends in a top-level domain.

Gigabit per second

Gigabit per second (Gb/s) is a unit of data transfer rate equal to: 1,000 Megabits per second.

Graphical User Interface

A GUI is an interface that lets humans interact with electronic devices through graphical icons.

High Availability

High Availability is a mode that permits the CMC to replicate its own data on another CMC.

Hypertext Transfer Protocol

HTTP is an application layer protocol in the Internet protocol suite model for distributed, collaborative, hypermedia information systems. HTTP is the foundation of data communication for the World Wide Web, where hypertext documents include hyperlinks to other resources that the user can easily access, for example by a mouse click or by tapping the screen in a web browser.

Hypertext Transfer Protocol Secure

HTTPS is an extension of the Hypertext Transfer Protocol (HTTP). It is used for secure communication over a computer network, and is widely used on the Internet. In HTTPS, the communication protocol is encrypted using Transport Layer Security (TLS) or, formerly, Secure Sockets Layer (SSL). The protocol is therefore also referred to as HTTP over TLS, or HTTP over SSL.

Identifier

A label that identifies the related item.

Industrial Control Systems

An ICS is an electronic control system and related instrumentation that is used to control industrial processes.

Information Object Addresses

An Information Object Address (IOA) is a unique identifier used in IEC 60870-5-101 and IEC 60870-5-104 protocols to specify individual data points, such as sensors or actuators, within a telecontrol system. The IOA format can vary and is configurable, typically appearing as either a 1-byte–2-byte or a 1-byte–1-byte–1-byte structure, depending on the system requirements.

Internet Assigned Numbers Authority

IANA is a standards organization that oversees global IP address allocation, root zone management in the Domain Name System (DNS), autonomous system number allocation, media types, and other Internet Protocol–related symbols and Internet numbers.

Internet Control Message Protocol

ICMP is a supporting protocol in the internet protocol suite. Network devices use it to send error messages and operational information to indicate success or failure when communicating with another IP address. ICMP differs from transport protocols such as TCP and UDP in that it is not typically used to exchange data between systems.

Internet of Things

The IoT describes devices that connect and exchange information through the internet or other communication devices.

Internet Protocol

An Internet Protocol address, or IP address, identifies a node in a computer network that uses the Internet Protocol to communicate. The IP label is numerical.

Intrusion Detection System

An intrusion detection system (IDS), which can also be known as an intrusion prevention system (IPS) is a software application, or a device, that monitors a computer network, or system, for malicious activity or policy violations. Such intrusion activities, or violations, are typically reported either to a system administrator, or collected centrally by a security information and event management (SIEM) system.

IPv6

Internet Protocol version 6 is the most recent version of the Internet Protocol, designed to replace IPv4. It uses 128-bit numerical addresses to identify devices, allowing for a vastly larger address space.

JavaScript Object Notation

JSON is an open standard file format for data interchange. It uses human-readable text to store and transmit data objects, which consist of attribute–value pairs and arrays.

Joint Photographic Experts Group

JPEG, or JPG, is a method of lossy compression that is used for digital images. The degree of compression can be adjusted, allowing a selectable tradeoff between storage size and image quality.

Kilobit per second

Kilobit per second (kb/s) is a unit of data transfer rate equal to: 1,000 bits per second.

Media Access Control

A MAC address is a unique identifier for a network interface controller (NIC). It is used as a network address in network segment communications. A common use is in most IEEE 802 networking technologies, such as Bluetooth, Ethernet, and Wi-Fi. MAC addresses are most commonly assigned by device manufacturers and are also referred to as a hardware address, or physical address. A MAC address normally includes a manufacturer's organizationally unique identifier (OUI). It can be stored in hardware, such as the card's read-only memory, or by a firmware mechanism.

Megabit per second

Megabit per second (Mb/s) is a unit of data transfer rate equal to: 1,000 kilobits per second.

Megabyte

The megabyte is a multiple of the unit byte for digital information. One megabyte is one million bytes.

National Institute of Standards and Technology

NIST is an agency of the United States Department of Commerce. NIST's mission is to promote American innovation and industrial competitiveness.

Network Interface Controller

A network interface controller (NIC), sometimes known as a network interface card, is a computer hardware component that lets a computer connect to a computer network.

Network Time Protocol

The NTP is a networking protocol to synchronize clocks between computer systems over variable-latency, packet-switched data networks.

Nozomi Networks Operating System

N2OS is the operating system that the core suite of Nozomi Networks products runs on.

Open Virtual Appliance

An OVA file is an open virtualization format (OVF) directory that is saved as an archive using the .tar archiving format. It contains files for distribution of software that runs on a virtual machine. An OVA package contains a .ovf descriptor file, certificate files, an optional .mf file along with other related files.

Operating System

An operating system is computer system software that is used to manage computer hardware, software resources, and provide common services for computer programs.

Operational Technology

OT is the software and hardware that controls and/or monitors industrial assets, devices and processes.

Portable Document Format

PDF is a Adobe file format that is used to present documents. It is independent of operating systems (OS), application software, hardware.

Privacy-Enhanced Mail

PEM is a standard file format that is used to store and send cryptographic keys, certificates, and other data. It is based on a set of 1993 IETF standards.

Programmable Logic Controller

A PLC is a ruggedized, industrial computer used in industrial and manufacturing processes.

Protected Extensible Authentication Protocol

PEAP is a protocol that encloses the Extensible Authentication Protocol (EAP) within an encrypted and authenticated Transport Layer Security (TLS) tunnel.

Random-access Memory

Computer memory that can be read and changed in any order. It is typically used to store machine code or working data.

Secure Shell

A cryptographic network protocol that let you operate network services securely over an unsecured network. It is commonly used for command-line execution and remote login applications.

Secure Sockets Layer

A secure sockets layer ensures secure communication between a client computer and a server.

Security Information and Event Management

SIEM is a field within the computer security industry, where software products and services combine security event management (SEM) and security information management (SIM). SIEMs provide real-time analysis of security alerts.

Server Message Block

Is a communication protocol which provides shared access to files and printers across nodes on a network of systems. It also provides an authenticated interprocess communication (IPC) mechanism.

Simple Network Management Protocol

SNMP is an Internet Standard protocol for the collection and organization of information about managed devices on IP networks. It also lets you modify that information to change device behavior. Typical devices that support SNMP are: printers, workstations, cable modems, switches, routers, and servers.

Simple Text Oriented Messaging Protocol

STOMP is a simple text-based protocol, for working with message-oriented middleware (MOM). It provides an interoperable wire format that allows STOMP clients to talk with any message broker supporting the protocol.

Text-based User Interface

In computing, a text-based (or terminal) user interfaces (TUI) is a retronym that describes a type of user interface (UI). These were common as an early method of human–computer interaction, before the more modern graphical user interfaces (GUIs) were introduced. Similar to GUIs, they might use the entire screen area and accept mouse and other inputs.

Thick Provisioning (Static)

Thick provisioning, also known as fat provisioning, is a method of allocating a fixed amount of physical storage to a virtual disk at creation. The full storage space is reserved immediately, regardless of how much data is actually stored.

Thin Provisioning (Dynamic)

Thin provisioning is a method of allocating storage space on a just-in-time basis as data is written, rather than reserving the full amount upfront. This approach makes more efficient use of available storage capacity.

Threat Intelligence™

Nozomi Networks **Threat Intelligence™** feature monitors ongoing OT and IoT threat and vulnerability intelligence to improve malware anomaly detection. This includes managing packet rules, YARA rules, STIX indicators, Sigma rules, and vulnerabilities. **Threat Intelligence™** allows new content to be added, edited, and deleted, and existing content to be enabled or disabled.

Transmission Control Protocol

One of the main protocols of the Internet protocol suite.

Transport Layer Security

TLS is a cryptographic protocol that provides communications security over a computer network. The protocol is widely used in applications such as: HTTPS, voice over IP, instant messaging, and email.

Uniform Resource Locator

An URL is a reference to a resource on the web that gives its location on a computer network and a mechanism to retrieving it.

Uninterruptible Power Supply

A UPS is an electric power system that provides continuous power. When the main input power source fails, an automated backup system continues to supply power.

Universal Serial Bus

Universal Serial Bus (USB) is a standard that sets specifications for protocols, connectors, and cables for communication and connection between computers and peripheral devices.

User Interface

An interface that lets humans interact with machines.

Variable

In the context of control systems, a variable can refer to process values that change over time. These can be temperature, speed, pressure etc.

Virtual Hard Disk

VHD is a file format that represents a virtual hard disk drive (HDD). They can contain what is found on a physical HDD, such as disk partitions and a file system, which in turn can contain files and folders. They are normally used as the hard disk of a virtual machine (VM). They are the native file format for Microsoft's hypervisor (virtual machine system), Hyper-V.

Virtual Machine

A VM is the emulation or virtualization of a computer system. VMs are based on computer architectures and provide the functionality of a physical computer.

Zettabyte File System

A file system and logical volume manager that combines storage pooling with data integrity features. N2OS uses ZFS for the `/data` file system.