Nagios XI – How to Integrate SNMP Traps With Nagios XI

Nagios®

The Industry Standard in IT Infrastructure Monitoring

Purpose

This document describes how to configure Nagios® XI™ to receive and process SNMP traps from external devices. Monitoring SNMP traps allows system administrators to monitor real-time events and network incidents in order to ensure an accurate and healthy monitoring environment.

Target Audience

This document is intended for use by Nagios administrators looking to integrate SNMP traps into their monitoring configuration to gain greater insight into their IT infrastructure.

Automated Installation

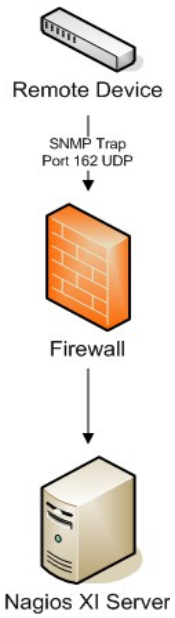
Open a terminal and login to the Nagios XI server as the root user and run the following commands:

```
cd /tmp
wget https://assets.nagios.com/downloads/nagiosxi/scripts/NagiosXI-SNMPTrap-setup.sh
sh ./NagiosXI-SNMPTrap-setup.sh
```

The `NagiosXI-SNMPTrap-setup.sh` script will do the following:

- Install all of the required prerequisites
- Download and install supporting files
- Modify the `snmptt.ini` and `snmptrapd.conf` files
- Add the `snmptt` user to the `nagios` and `nagcmd` groups
- Modify some permissions
- Add a firewall rule in iptables to open UDP port 162
- Set up the `snmptt` and `snmptrapd` daemons to start automatically on boot

Intermediary Firewalls



Before you can configure remote devices to send SNMP traps to Nagios XI you will have to configure any intermediary firewalls between the Nagios XI server and the remote device to allow inbound SNMP traps to be sent to Nagios XI. This involves allowing UDP port 162 traffic from remote devices to the Nagios XI server.

Remember that unlike with most checks, Nagios XI is the server (rather than the client) for SNMP traps, so the packet flow is inbound to the Nagios XI machine.

A firewall rule was added to iptables to open UDP port 162 (by the script `NagiosXI-SNMPTrap-setup.sh`) which you ran during the Automated Installation section above.

Installing MIBs

You may need to configure **snmptt** on the Nagios XI server to use the MIBs your remote devices are using. This can be done via **Admin** > **System Extensions** > **Manage MIBs**.

Click the Browse button to find the MIB to be added. Check the box **Process trap** and then click the **Upload MIB** button.

This will find any trap definitions in the MIB file and add them to the `/etc/snmp/snmptt.conf` file, they will be added as an EVENT to this file.

The MIB will also be copied into the `/usr/share/snmp/mibs/` directory.

If you had previously uploaded a MIB file but did not select the **Process trap** check box you can run the following command in a terminal session:

```
addmib <PathToNewMIB>
```

For example:

```
addmib /usr/share/snmp/mibs/NAGIOS-NOTIFY-MIB.txt
```

SNMPTT

SNMP Trap Translator is what processes the received traps and decides if they should be sent to Nagios XI. This documentation will briefly explain how SNMPTT works and can be configured.

EVENT / FORMAT / EXEC

Trap definitions are defined in the `/etc/snmp/`**`snmptt.conf`** file, they always start with **EVENT** line are followed by a **FORMAT** line and an **EXEC** line. There are other lines that will exist but are not mandatory. For example anything between **SDESC** and **EDESC** is purely comment information and is not processed as part of the trap.

EVENT

This is the line that has the OID / MIB, if this is matched against the incoming trap then SNMPTT will action it by executing the EXEC line.

FORMAT

This allows you to define what is logged in the `/var/log/snmptt/snmptt.log` file when an EVENT is matched. If a received trap is not matched by SNMPTT then it will be logged in the `/var/log/snmptt/snmpttunknown.log` file.

EXEC

This is the line that submits the received trap to Nagios XI. By default it will execute the `/usr/local/bin/snmptraphandling.py` script which will submit the check result to Nagios XI.

Here is an example:

```
EVENT linkDown .1.3.6.1.6.3.1.1.5.3 "Status Events" Critical
FORMAT Link down on interface $1.  Admin state: $2.  Operational state: $3
EXEC /usr/local/bin/snmptraphandling.py "$r" "SNMP Traps" "$s" "$@" "$-*" "Link down on
interface $1.  Admin state: $2.  Operational state: $3"
```

Very briefly:

The EVENT line will define this trap as CRITICAL when submitted to Nagios XI

The EXEC line will be targeting the service called **SNMP Traps** when submitted to Nagios XI

Note: If you find the EXEC line does not look like the examples above, please follow the steps in the following KB article:

https://support.nagios.com/kb/article.php?id=559

SNMPTT Processing Behavior

It's important to note that SNMPTT will compare a received trap against every EVENT in the `snmptt.conf` file. Multiple EVENTS with the same OID / MIB can exist and hence multiple EXEC statements can be executed. The key point being made here is that SNMPTT does not stop looking through the `snmptt.conf` file once an EVENT is matched.

This behavior allows for more complicated EVENTs that have filters applied using MATCH lines. If you have two identical EVENTS, it's possible that you'll submit two traps to Nagios XI at the same time and the last trap received will overwrite the previous one. This is outside of the scope of this documentation however it is worth mentioning.

Send Test SNMP Trap

It's very easy to send a test trap to Nagios XI to demonstrate how traps are received by Nagios XI.

In a terminal session execute the following command:

```
snmptrap -v 2c -c public 127.0.0.1 '' linkUp ifDescr s eth0 ifAdminStatus i 1 ifOperStatus i 1
```

Once you execute this command, you'll see the following logged in the `/var/log/snmptt/snmptt.log` file:

```
Mon Nov 28 11:15:42 2016 .1.3.6.1.6.3.1.1.5.4 Normal "Status Events" localhost - Link up on
interface eth0.  Admin state: up.  Operational state: up
```

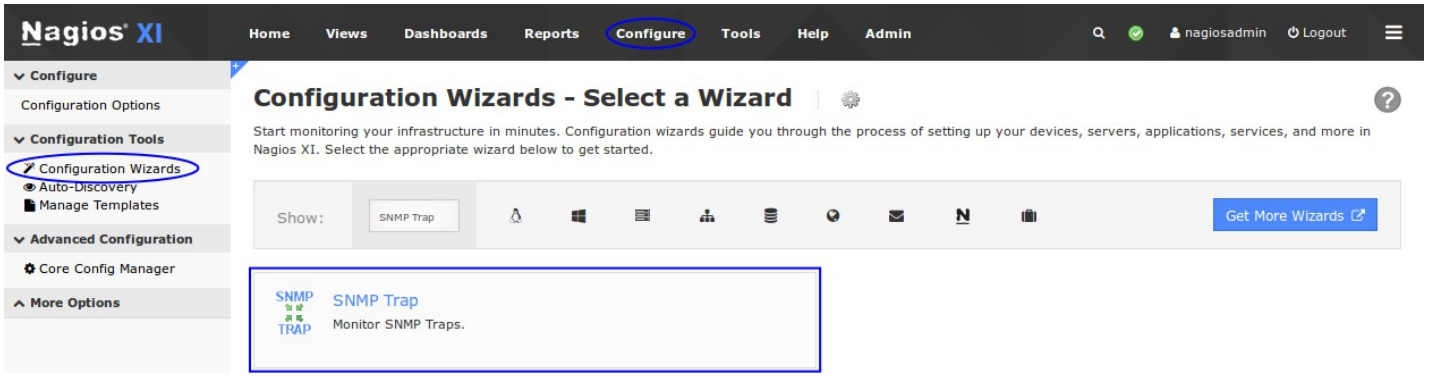
You'll also see the following logged in the `/usr/local/nagios/var/nagios.log` file:

```
[1480298939] Warning:  Passive check result was received for service 'SNMP Traps' on host
'localhost', but the service could not be found!
[1480298939] Error: External command failed -> PROCESS_SERVICE_CHECK_RESULT;localhost;SNMP
Traps;0;Link up on interface eth0.  Admin state: up.  Operational state: up / ifDescr
(OCTETSTR):eth0 ifAdminStatus (INTEGER):up ifOperStatus (INTEGER):up
```

Nagios XI has now received the SNMP Trap however as you can see from the message above it is reporting that the **SNMP Traps** service could not be found. The next step will show you how to use the SNMP Trap wizard to create this service in Nagios XI.

Using The SNMP Trap Wizard

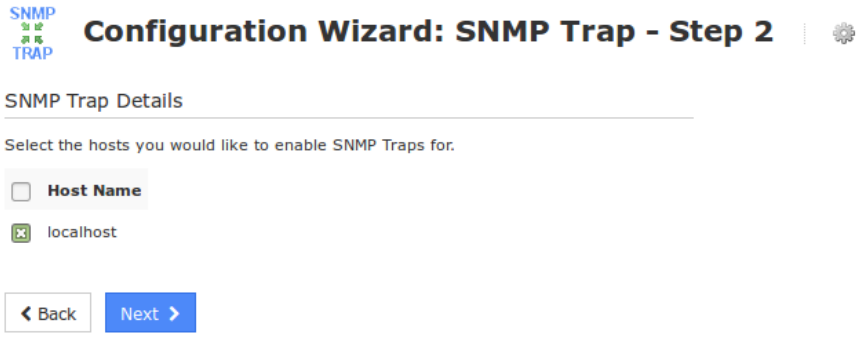
Each host or device that you wish to receive and process SNMP traps for must have a corresponding SNMP Traps service defined in Nagios XI. Nagios XI has a built-in wizard that makes the configuration of these SNMP trap events quick and simple. Navigate via the top menu bar to **Configure** > **Run a configuring wizard**, and select the **SNMP Trap** wizard. In the following screenshot you can see how the search field allows you to quickly find a wizard.



The first screen says **This wizard allows you to enable SNMP Traps for existing hosts that are being monitored**. Click Next to continue.

The wizard will then ask you which host you wish to add an SNMP trap service.

When you have selected all the hosts you want click **Next**.



Complete the wizard by choosing the required options in Step 3 – Step 5. To finish up, click on **Finish** in the final step of the wizard. This will create the new service called **SNMP Traps** and will be waiting to receive a trap.

Once the wizard applies the configuration, click the **View status details for xxxxx** link to see the new service that was created.

Host	Service	Status	Duration	Attempt	Last Check	Status Information
localhost	SNMP Traps	Ok	37s	1/1	2016-11-28 13:37:14	Waiting for trap...

Send Test SNMP Trap

By sending a test trap you'll be able to see how it's received in XI. In a terminal session execute the following command:

```
snmptrap -v 2c -c public 127.0.0.1 '' linkUp ifDescr s eth0 ifAdminStatus i 1 ifOperStatus i 1
```

Once you execute this command, you'll see the service update as follows:

Host	Service	Status	Duration	Attempt	Last Check	Status Information
localhost	SNMP Traps	Ok	33s	1/1	2016-11-28 13:44:32	Link up on interface eth0. Admin state: up. Operational state: up / ifDescr (OCTETSTR):eth0 ifAdminStatus (INTEGER):up ifOperStatus (INTEGER):up

Being a linkUp trap the service was submitted with an OK status. This is because the EVENT line ends with **Normal** (OK in Nagios XI) and the **EXEC** line sends the Normal state with the **"$s"** variable:

```
EVENT linkUp .1.3.6.1.6.3.1.1.5.4 "Status Events" Normal
FORMAT Link up on interface $1.  Admin state: $2.  Operational state: $3
EXEC /usr/local/bin/snmptraphandling.py "$r" "SNMP Traps" "$s" "$@" "$-*" "Link up on interface $1.
Admin state: $2.  Operational state: $3"
```

Now send a test trap for a linkDown and you'll be able to see how it's received in XI. In a terminal session execute the following command:

```
snmptrap -v 2c -c public 127.0.0.1 '' linkDown ifDescr s eth0 ifAdminStatus I 2 ifOperStatus I 2
```

Once you execute this command, you'll see the service update as follows:

Host	Service	Status	Duration	Attempt	Last Check	Status Information
localhost	SNMP Traps	Critical	5s	1/1	2016-11-28 13:54:50	Link down on interface eth0. Admin state: down. Operational state: down / ifDescr (OCTETSTR):eth0 ifAdminStatus (INTEGER):down ifOperStatus (INTEGER):down

Being a linkDown trap the service was submitted with a CRITICAL status. This is because the EVENT line ends with **Critical** and the **EXEC** line sends the CRITICAL state with the **"$s"** variable:

```
EVENT linkDown .1.3.6.1.6.3.1.1.5.3 "Status Events" Critical
FORMAT Link down on interface $1.  Admin state: $2.  Operational state: $3
EXEC /usr/local/bin/snmptraphandling.py "$r" "SNMP Traps" "$s" "$@" "$-*" "Link down on interface
$1.  Admin state: $2.  Operational state: $3"
```

The SNMP Traps service will stay in a CRITICAL state until the next trap is received.

SNMP Traps Are Passive

An important point to stress with SNMP traps is that they are asynchronous events that can occur at any time, in Nagios XI this is what is called a **PASSIVE** check/service. This means that they are not actively checked by Nagios XI on a regular schedule, Nagios XI is waiting for a SNMP Trap to be received from the remote device. A comparison between an active check and a passive check helps explain the difference between **ACTIVE** and **PASSIVE** checks:

> UPS device loses input power and is running on batteries.
>
> > With an **active** check, if Nagios XI was checking the device on a 5 minute interval then it might be up to 5 minutes before Nagios XI is aware that the device is on batteries.
> >
> > With a **passive** check, the device immediately sends an SNMP Trap to Nagios XI when it is running on batteries.

More detailed information on passive service can be found in the following documentation:

> https://assets.nagios.com/downloads/nagiosxi/docs/Configuring-Passive-Services-With-Nagios-XI.pdf

Troubleshooting

SNMP traps can get very complicated and generally require some knowledge and troubleshooting to get working just the way you want. Here is an outline of a general troubleshooting for SNMP traps. Please note that if you are attempting to use this troubleshooting guide without using the above install script, your battle will be uphill as the script enables various aspects of **SNMPTT** that we will use exhaustively.

First thing that is helpful is a separate server that we can send test traps from, this can also be done from the Nagios XI server although it will not validate any firewall rules that may be in place. Sending a test trap using the `snmptrap` command used in the previous examples will send a valid trap to the Nagios XI server, for example:

```
snmptrap -v 2c -c public <NAGIOS XI SERVER IP> '' netSnmpExampleHeartbeatNotification
netSnmpExampleHeartbeatRate I 123456
```

This will send an SNMP trap to your Nagios XI server. Remember to replace <**NAGIOS XI SERVER IP**> with the IP address of your Nagios XI server.

Now that you've sent the test trap, you should check a few things to make sure its all working. The specific trap that was sent DOES NOT exist in the `snmptt.conf` file, hence it will be logged in the file:

```
/var/log/snmptt/snmpttunknown.log
```

There should be logs of your test SNMP trap here (at the bottom of the file). If there is not, make sure that there is not some intermediary firewall in the way. Check to make sure iptables is allowing traffic through on port 162. Do not progress past this point until you are able to get this test trap. The following KB articles provide more detailed troubleshooting steps:

SNMP Trap - Inbound UDP Traffic

https://support.nagios.com/kb/article.php?id=86

SNMP Trap - Firewall Rules

https://support.nagios.com/kb/article.php?id=87

If you are able to receive a trap, you are ready to start capturing real SNMP traps. Monitor `/var/log/snmptt/snmptt.log` for SNMP traps that are coming in. Also make sure that traps are not getting relegated to unknown status by keeping an eye on `snmpttunknown.log`.

If you are seeing traps in your `/var/log/snmptt/snmptt.log` but cannot locate them within your Nagios XI system, it may be that you have
not set up your SNMP Traps service for the remote host sending the traps. Nagios XI is receiving these traps however is discarding the results as there is no service defined for the host that the trap belong to.

Nagios XI has a section called **Unconfigured Objects** which allows you to see the passive checks that have been received by Nagios XI, but no object exists for them. Navigate within the XI web-interface to **Admin** > **Monitoring Config** > **Unconfigured Objects**. You can either set up the SNMP Traps service using the **SNMP Traps** wizard (demonstrated above) OR by clicking on the blue triangle under actions which runs the **Unconfigured Passive Object** wizard. Further information on the **Unconfigured Passive Object** wizard can be found in the following documentation:

https://assets.nagios.com/downloads/nagiosxi/docs/Monitoring_Unconfigured_Objects_With_XI.pdf

The following KB articles may also help with your troubleshooting:

SNMP Trap - snmptrapd Service

https://support.nagios.com/kb/article.php?id=88

SNMP Trap - snmpttt Service

https://support.nagios.com/kb/article.php?id=89

Further Reading

More detailed examples for sending test SNMP Traps can be found in the following KB article:

SNMP Trap - How To Send A Test Trap

https://support.nagios.com/kb/article.php?id=493

The following tutorial goes into extensive detail to explain how SNMP Traps work in Nagios XI and explain how to setup a test environment:

Nagios XI - SNMP Trap Tutorial

https://support.nagios.com/kb/article.php?id=77

If you are having difficulties with SNMP Traps and IPv6 please read the following KB article:

Nagios XI - Receiving IPv6 SNMP Traps

https://support.nagios.com/kb/article.php?id=499

Information on the variables in SNMP Traps:

SNMP Traps - Understanding Trap Variables

https://support.nagios.com/kb/article.php?id=558

Standard Handler vs Embedded Handler:

SNMP Traps - Standard Handler vs Embedded Handler

https://support.nagios.com/kb/article.php?id=557

SNMPTT documentation including the format of the `snmptt.conf` file:

http://snmptt.sourceforge.net/docs/snmptt.shtml

For more information on OIDs and what a given number is for, see http://www.oid-info.com/. You are encouraged to submit descriptions for any OIDs you know that are not in the repository yet. Not all event names will be as obvious as linkDown, so you may need to do some research to figure out what to use in your configuration. The MIBs you use may come with documentation that describes what event names can be used.

Final Thoughts

SNMP traps are a great method for monitoring asynchronous events in your IT infrastructure. The complexity of managing MIBs and the intricacies of the SNMP protocol can often be daunting, but if you get familiar with the in and outs of SNMP, it can be a powerful addition to your IT infrastructure management and allow for advanced, real-time network event monitoring.

For any support related questions please visit the Nagios Support Forums at:

https://support.nagios.com/forum/