Using the NSClient for Passive Checks With Nagios XI

Purpose

This document describes how to configure NSClient++ to send passive check results to Nagios XI using Nagios Remote Data Processor (NRDP) Nagios Service Check Acceptor (NSCA).

Check results received from external devices / applications is what defines a Passive check. It's the responsibility of the external devices / applications to send the check results through, all Nagios XI does is wait for the results (as opposed to Active checks where Nagios XI is responsible for performing the check on a schedule). Passive checks reduce the load on your Nagios XI server by reducing the number of active checks run. Passive checks are also useful for security-related and asynchronous events you wish to monitor.

NRDP and NSCA are both listeners that receive the passive check results and submit them to Nagios XI. Historically NSCA was the only method available however NRDP is now recommended as it simply runs as a web page on the Nagios XI server and is less complicated to setup. This documentation provides steps to configure NSClient++ to send to either listener.

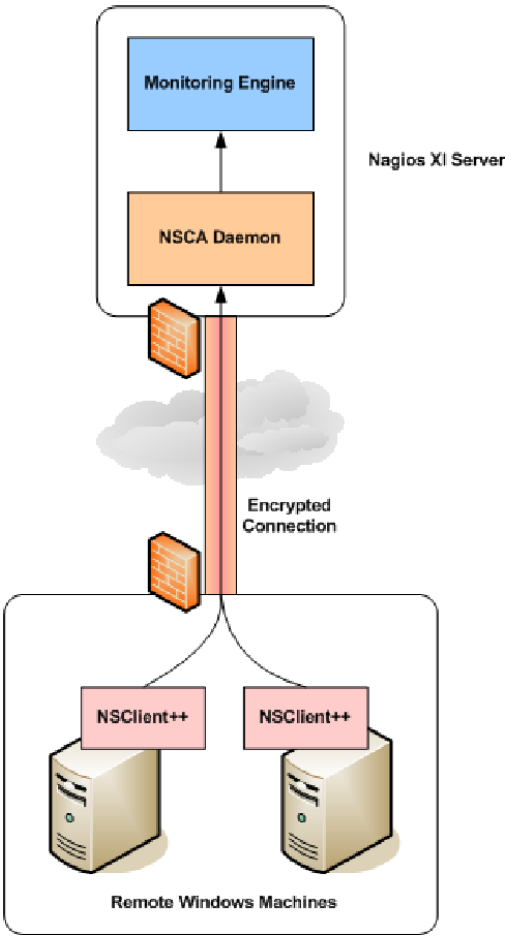
Solution Overview

Configuring NSClient++ to work as a passive monitoring agent for remote Windows machines is ideal for managed service providers and organizations with remote offices and roaming laptops.

Configuring the agent to operate in passive mode allows for a simple method of monitoring Windows services and metrics behind firewalls, proxies, and private address spaces. Under most circumstances, no configuration or alteration of remote firewalls is required to implement this type of monitoring.

Firewall rules may be required to allow NSClient++ to send the outbound check results to the Nagios XI server. The central NOC or datacenter where the Nagios XI server is located will also need firewall rules to allow the inbound check results to the Nagios XI server. The network ports are as follows:

- **NRDP** = TCP port 80 or 443
- **NSCA** = TCP port 5667



Data communication between remote agents and the Nagios server may be encrypted to ensure secure transfers.

- **NRDP** = Encryption is achieved when using HTTPS over port 443 using SSL/TLS certificates
- **NSCA** = Encryption is configured on NSClient++ and the NSCA listener on Nagios XI

With passive checks, it's the responsibility of NSClient++ to send the check results through to Nagios XI. NSClient++ uses a scheduler to execute the checks on a regular basis, hence all the checks that need to be executed are defined in the NSClient++ configuration file.

Prerequisites

Before you continue following the instructions outlined in this document, you must have met the following prerequisites.

- The NDRP or NSCA listener must be configured properly on the Nagios XI server
  - Please refer to the following documentation on how to configure NRDP or NSCA on Nagios XI.
- If using NRDP:
  - It is recommended to implement SSL/TLS as per the following documentation.
  - You need to use a more recent version of NSClient++ as it does not work in earlier versions. This guide was tested with version 0.5.1.29.
- NSClient++ 0.4.x or newer must be installed on the Windows machines being monitored. Please refer to this documentation on how to install NSClient++.
  - The older version 0.3.x of NSClient++ is not compatible with the newer version of **NSCA** that comes with Nagios XI, hence this guide only focuses on configuring NSClient++ 0.4.x+.
  - If you are using **NRDP** you need to use the very latest version of NSClient++ as it does not work in earlier versions.
  - This guide was tested with version 0.5.1.29
    - https://github.com/mickem/nscp/releases

Agent Configuration

You must modify the NSClient++ configuration to support sending passive check results to the Nagios XI server. The configuration file in NSClient++ 0.4.x+ is called `nsclient.ini` and is located here:

```
C:\Program Files\NSClient++\nsclient.ini
```

The following steps will require you to execute some commands in a command prompt to load the modules in the `nsclient.ini` file. These commands also add all the default options for the `nsclient.ini` which makes it easier to configure. After executing these commands, the nsclient.ini file will be manually edited to finalize the settings to communicate with Nagios XI.

Log onto your remote windows machine as an administrator.

Open a command prompt with administrative rights and run the following commands:

```
cd "C:\Program Files\NSClient++"
nscp settings --activate-module Scheduler --add-defaults
nscp settings --activate-module CheckSystem --add-defaults
nscp settings --activate-module CheckDisk --add-defaults
nscp settings --activate-module CheckHelpers --add-
```

This should not produce any output, don't be alarmed if you see some “Failed to register plugin” errors.

NRDP Only:

```
nscp settings --activate-module NRDPClient --add-defaults
```

NSCA Only:

```
nscp settings --activate-module NSCAClient --add-defaults
```

Open the configuration file `nsclient.ini` in a text editor for editing (like Notepad) to continue configuring NSClient++.

Under the [/modules] section, confirm following lines exist:

```
NSCAClient OR NRDPClient = enabled
Scheduler              = enabled
CheckSystem            = enabled
CheckDisk               = enabled
CheckHelpers           = enabled
```

Next you need to configure the settings that define the Nagios XI server this agent will be sending passive check results to. The following settings are different for NRDP and NSCA and are explained separately.

NRDP

Find the section **[/settings/NRDP/client/targets/default]** and remove the existing lines except for the timeout line. Define the following lines:

```
address = https://10.25.5.13/nrdp/
token = LIqdtq0Y7fPU
```

The address is the URL of NRDP on your Nagios XI server and is normally `https://xxxxxxx/nrdp/`.

If you implemented SSL/TLS with Nagios XI the address needs to start with https.

If you do not implement SSL/TLS then the traffic will NOT be encrypted, you will use http instead.

The token will have been defined on your Nagios XI server NRDP settings via **Admin** > **Check Transfers** > **Inbound Transfers** > **NRDP**.

Proceed to the Hostname Setting section in this document.

NSCA

Find the section **[/settings/NSCA/client/targets/default]** and define the following (add them if they do not exist):

```
address = 10.25.5.13
encryption = 3
password = Str0ngPassw0rd
```

The address is the IP address OR DNS record of your Nagios XI server.

The settings encryption and password need to match the settings that have been defined on your Nagios XI server NSCA settings. These would have been configured via **Admin** > **Check Transfers** > **Inbound Transfers** > **NSCA**. For clarity you can refer to the file `usr/local/nagios/etc/nsca.cfg` on the Nagios XI server to verify the settings that are being used. In this example we are using 3DES which is the number 3 in `nsclient.ini`.

Proceed to the Hostname Setting section in this document.

Hostname Setting

The following setting applies to both NRDP and NSCA, it is just in a different location in the nsclient.ini file. Find the section:

- NRDP

  [/settings/NRDP/client]

- NSCA

  [/settings/NSCA/client]

Observe the following setting:

```
hostname = auto
```

This is how the **name** of this Windows computer will be identified on the Nagios XI server when the passive check results are received. There are some other options available such as:

```
auto            =    Hostname

${host}         =    Hostname

${host_lc}      =    Hostname in lowercase

${host_uc}      =    Hostname in uppercase

${domain}       =    Domainname

${domain_lc}    =    Domainname in lowercase

${domain_uc}    =    Domainname in uppercase
```

You can also just type the name such as **hostname = My Windows Machine** however this would be tedious to implement on multiple clients.

Most likely you have some sort of naming standard in Nagios XI, you could use the following:

```
hostname = ${host_uc}
```

This would target the hostname WIN7-04. Another example could be:

```
hostname = ${host_lc}.${domain_lc}
```

This would target the hostname `win7-04.box293.local`.

Proceed to the Checks section in this document.

Checks

Now to define the checks that will be executed and sent to Nagios XI. The same setting applies to both NRDP and NSCA. Find the section **[/settings/scheduler/schedules]** and add the following lines:

```
host_check              = Check_OK "Everything is working fine"
CPU Load                = checkCPU warn=80 crit=90 time=5m time=1m time=30s
Drive C: Disk Usage     = CheckDriveSize MinWarn=10% MinCrit=5% Drive=C: ShowAll
Uptime                  = CheckUptime MinCrit=12h ShowAll
Print Spooler Service   = check_service service=spooler
Memory Usage            = checkMem MaxWarn=80% MaxCrit=90% ShowAll=long
type=physical type=virtual type=paged type=page
```

For clarity, the **Memory Usage** is shown as the last line in this example due to the length of the command.

The lines start with the name of the service that will be identified on the Nagios XI server when the passive check results are received, for example `Drive C: Disk Usage`.

Proceed to the Schedule section in this document.

Schedule

Finally, you need to define how frequently you want the checks on NSClient++ to be executed and sent back to Nagios XI. You will need to add the following section as it will not exist yet:

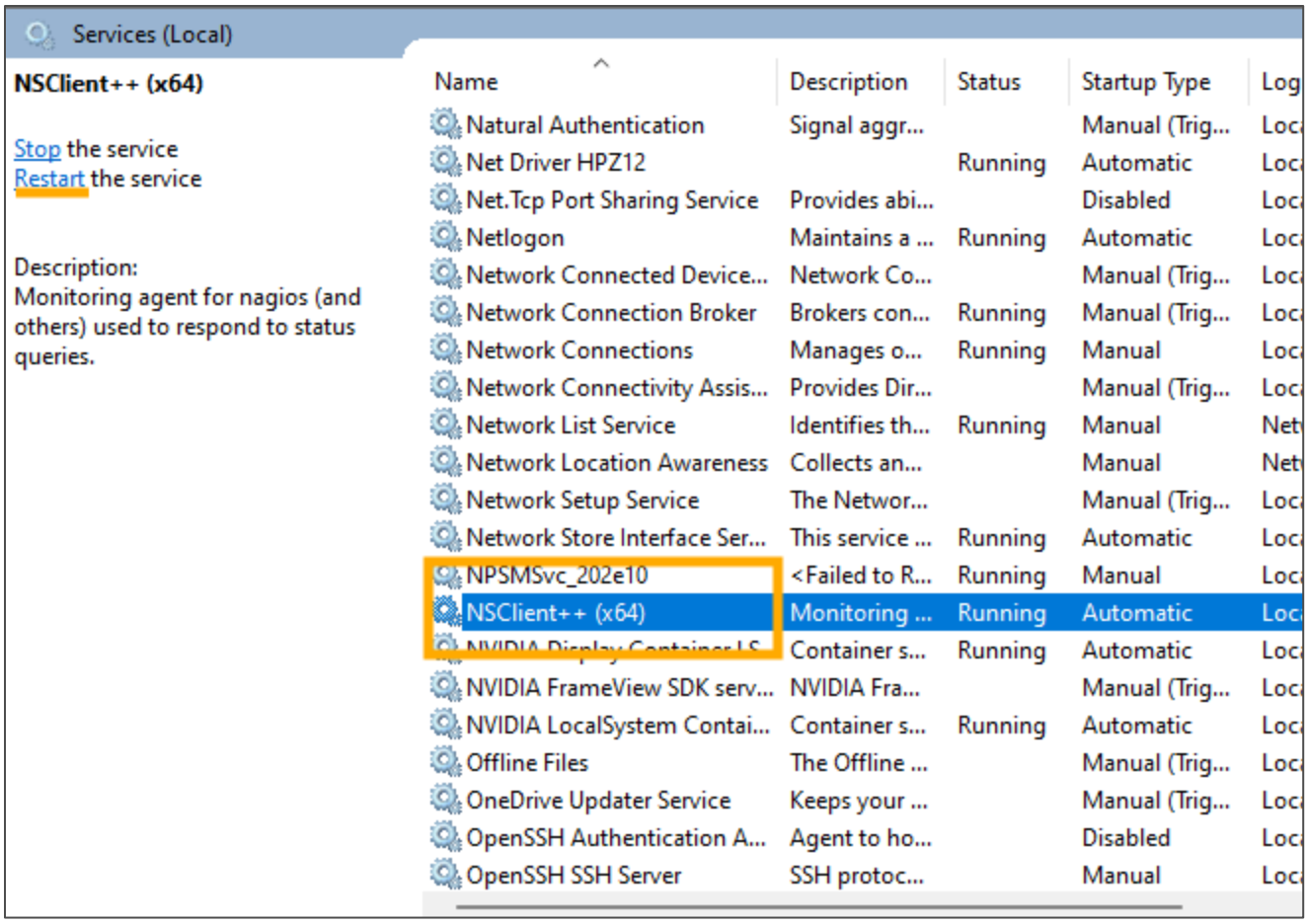
```
[/settings/scheduler/schedules/default]
interval = 5m
channel = NRDP OR NSCA
```

The channel line determines if NSClient++ uses NRDP or NSCA.

This completes the configuration of the nsclient.ini file, please save the changes. In the next section you will be shown how to restart the NSClient++ service which will implement the changes.

Restarting The NSClient++ Service

NSClient must now be restarted. In Windows open the **Services** console under **Administrative Tools**. If you cannot locate this, use `services.msc` to open the **Services** console.



Locate the **NSClient++** service, right click it and select **Restart**. You can close the **Services** console as it's no longer required.

Nagios XI Configuration

Once you configure the agent, it will start transmitting passive check results to the Nagios XI server.

As Nagios XI has not yet been configured to process results from the remote machine, the remote machine's status information will not appear in the interface. To add the remote machine's information to the monitoring engine, you'll need to configure monitoring of the newly arriving passive checks by using Nagios XI's Unconfigured Objects feature.

Information on using the Unconfigured Objects feature can be found in the following documentation:

What Can I Monitor With NSClient++

Any of the available modules in NSClient++ can be used to create services under the **[/settings/scheduler/schedules]** section including external scripts. Please refer to the official documentation to see what modules are available:

Troubleshooting

There are several log files you can look at to investigate issues you may be having.

Windows Computer

The log file `C:\Program Files\NSClient++\nsclient.log` is a good place to start your troubleshooting. You might need to enable debug logging with the following command:

```
nscp settings --path /settings/log --key level --set debug
```

Then restart the NSClient++ serviceNagios XI Server.

The `xinetd` daemon will log incoming connections to the `/var/log/messages` log file. You can watch this with the following command:

```
tail -f /var/log/messages
```

The nagios daemon will log passive check results for services that it does not know about in the `/usr/local/nagios/var/nagios.log` file. You can watch this with the following command:

```
tail -f /usr/local/nagios/var/nagios.log
```

The following KB articles will provide more detailed troubleshooting steps:

NSCA Server - Inbound TCP Traffic
NSCA Server - Firewall Rules
NSCA Server - Debug Logging

Finishing Up

This completes the documentation on configuring NSClient++ for passive checks with Nagios XI. If you have additional questions or other support-related questions, please visit us at our Nagios Support Forum, Nagios Knowledge Base, or Nagios Library:

Visit Nagios Support Forum Visit Nagios Knowledge Base Visit Nagios Library