



Purpose

This document explains how to configure the NSClient++ Windows agent and what the options in the configuration file mean.

Target Audience

This document is intended for use by Nagios Administrators and IT desktop support staff who wish to understand how the configuration file works for NSClient++ to enable some of the advanced features.

Agent Overview

NSClient++ must be installed on the target Windows machine and configured before Nagios XI can monitor system metrics, services, processes, or performance data on the target machine. If you haven't installed the NSClient++ agent on the remote machine, please, refer to our documentation on installing the XI Windows agent:

[Installing The Windows Agent NSClient++ For Nagios XI](#)

Once the NSClient++ agent has been installed following the guide above, the Windows desktop or server is ready to be monitored by using the **NSClient++** Wizard, for detailed information please refer to the following documentation:

[Monitoring Windows Using NSClient++](#)

This documentation applies to both the 0.3.x and 0.4.x versions of NSClient++:

- The last release of the 0.3.x branch was in July 2011 and the developer of NSClient++ no longer supports the 0.3.x version
- The 0.4.x and newer versions have a different configuration file format

With this in mind, the 0.4.x and newer versions are the primary focus of this document, however 0.3.x information is still present for those users who still have this version deployed in their environment.

Configuring NSClient++ 0.4.x

The configuration file in NSClient++ 0.4.x is called `nsclient.ini` and is located here:

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

This document is going to show you examples of the different options available in NSClient++. The NSClient++ installer creates the `nsclient.ini` file with the minimal settings that allow NSClient++ to work (*defined by your choices during the installation*). To use the options described in this document you will need to populate the `nsclient.ini` file with the additional options.

Open a Command Prompt as an Administrator and run the following commands:

```
cd "C:\Program Files\NSClient++"  
nscp settings --generate --add-defaults --load-all
```

This should not produce any output, however don't be alarmed if you see some "Failed to register plugin" messages.

At this point if you open `nsclient.ini` in a text editor you will see all the available options for the modules you have enabled.

NOTE: You will need to restart the NSClient++ service whenever you make changes, see the [Restarting The NSClient++ Service](#) section of this document for steps on how to do this.

Sections and Settings

The configuration file contains separate sections for the different configuration options that are enclosed in square brackets, for example:

```
[/settings/default]
password = Str0ngPassw0rd

[/modules]
NSClientServer = 1

[/settings/NSClient/server]
; PORT NUMBER - Port to use for check_nt.
port = 12489
```

- The lines enclosed by [square brackets] are the different sections in the configuration file. Items under that section are the specific settings items for that section
- The sections do not need to be in any special order
- Lines that begin with a semi-colon (;) are comments only
- It is important that you only use valid settings for the section they reside in, using incorrect settings can cause unexpected results

The following sections will explain their purpose and what options are available. Not all sections and options will be explained in this document, only the relevant options that are using by Nagios XI for monitoring. Some sections will redirect you to other documentation that provides more detailed information.

`[/modules]`

The `[/modules]` section is the most important section in the configuration file. The settings in this section define what modules of NSClient++ are enabled. You can enable/disable a module by setting its value to 1 or 0, for example:

```
NSClientServer = 1
CheckDisk = 1
CheckEventLog = 1
WebServer = 0
Scheduler = 1
```

If a module does not exist, you can enable a module and load all of its default settings by executing the following command (*CheckDisk used in the following example*):

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

`[/settings/default]`

There are some common options in different sections that can be defined here and hence do not need to be defined in each individual section. For example:

```
allowed hosts = 127.0.0.1,10.25.5.11,10.25.5.12
timeout = 30
```

These two settings would be inherited by the `[/settings/NSClient/server]` and `[/settings/NRPE/server]` sections. This saves having to define them in separate locations.

[/settings/log]

This section allows you to enable debug logging to help diagnose issues you may be having with NSClient++, for example:

```
level = debug
```

[/settings/NSClient/server]

This section will become available once the `NSClientServer` module has been loaded. Settings specific to the `check_nt` plugin can be defined here. The Nagios XI Configuration Wizard **NSClient++** creates services that use this module using the `check_nt` plugin.

If you wanted to change the default port you would change it with this setting:

```
port = 12489
```

This module is what is known as **active** checks, the Nagios XI server is actively performing the checking.

[/settings/NRPE/server]

This section will become available once the `NRPEServer` module has been loaded. This is explained in more detail in the following documentation:

[Enabling The NRPE Listener In NSClient 0.4.x For Nagios XI](#)

The NRPE module allows for more advanced monitoring of the Windows machine, compared to what is available using the `NSClientServer` module. For example:

- Execute custom scripts / plugins
- Customize the output of data returned from checks like disk usage, performance counters etc
- More advanced versions of the CPU, Memory and Disk checks
- This module is what is known as **active** checks, the Nagios XI server is actively performing the checking

More detailed information on all the options available using the `NRPEServer` module can be found in the official documentation:

<https://docs.nsclient.org/reference/index.html#windows-modules>

`[/settings/NRDP/client]` OR `[/settings/NSCA/client]`

These sections will become available once the `NRDPClient` or `NSCAClient` module has been loaded. This is explained in more detail in the following documentation:

[Using NSClient++ For Passive Checks](#)

- Using NRDP or NSCA means that it is up to the client to send the check results back to Nagios XI
- Good for roaming devices like laptops
- Ideal for remote devices that do not have a static public network address or are behind a firewall
- These modules are what is known as passive checks, the Nagios XI server is passively waiting for the client to submit check results.

`[/settings/external_scripts/scripts]`

This section will become available once the `CheckExternalScripts` module has been loaded. The `CheckExternalScripts` module is what allows custom scripts / plugins to be executed to provide more advanced monitoring of the Windows machine, it is utilized by the the `NRPEServer`, `NRDPClient` and `NSCAClient` modules.

The following KB article provided detailed examples on how to add your own scripts / plugins to this section:

<https://support.nagios.com/kb/article.php?id=528>

NSClient++ 0.4.x Summary

This has been a basic overview of how the NSClient++ 0.4.x configuration options work. For more information on configuring NSClient++ 0.4.x and newer please visit:

<https://docs.nsclient.org/reference/index.html#windows-modules>

Configuring NSClient++ 0.3.x

The configuration file in NSClient++ 0.3.x is called `NSC.ini` and is located here:

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

This document is going to show you examples of the different options available in NSClient++. The NSClient++ installer creates the `NSC.ini` file populated with all the available settings for NSClient++.

NOTE: You will need to restart the NSClient++ service whenever you make changes, see the [Restarting The NSClient++ Service](#) section of this document for steps on how to do this.

Sections and Settings

The configuration file contains separate sections for the different configuration options, for example:

```
[modules]
```

```
NRPEListener.dll  
NSClientListener.dll
```

```
[Settings]
```

```
use_file=1  
allowed_hosts=10.25.5.11,10.25.5.12
```

- The lines enclosed by [square brackets] are the different sections in the configuration file. Items under that section are the specific settings items for that section
- The sections do not need to be in any special order
- Lines that begin with a semi-colon (;) are comments only
- It is important that you only use valid settings for the section they reside in, using incorrect settings can cause unexpected results

The following sections will explain their purpose and what options are available. Not all sections and options will be explained in this document, only the relevant options that are using by Nagios XI for monitoring. Some sections will redirect you to other documentation that provides more detailed information.

[modules]

The settings in this section define what modules of NSClient++ are enabled. You can enable a module by removing the semi-colon ; or disable a module by adding the semi-colon ; to comment it. For example:

```
NRPEListener.dll
NSClientListener.dll
;CheckWMI.dll
CheckSystem.dll
```

[Settings]

There are some common options in different sections that can be defined here and hence do not need to be defined in each individual section. For example:

```
allowed hosts=127.0.0.1,10.25.5.11,10.25.5.12
timeout=30
```

These two settings would be inherited by the [NSClient] and [NRPE] sections. This saves having to define them in separate locations.

[log]

This section allows you to enable debug logging to help diagnose issues you may be having with NSClient++, for example:

```
debug=1
```


[NSClient]

This section will become available once the `NSClientListener.dll` module has been loaded. Settings specific to the `check_nt` plugin can be defined here. The Nagios XI Configuration Wizards **NSClient++** create services that use this module using the **check_nt** plugin.

If you wanted to change the default port you would change it with this setting:

```
port=12489
```

This module is what is known as **active** checks. The Nagios XI server is actively performing the checking.

[NRPE]

This section will become available once the `NRPEListener.dll` module has been loaded. This is explained in more detail in the following documentation:

[Enabling The NRPE Listener In NSClient++ 0.3.x](#)

The NRPE module allows for more advanced monitoring of the Windows machine, compared to what is available using the `NSClientListener.dll` module. For example:

- Execute custom scripts / plugins
- Customize the output of data returned from checks like disk usage, performance counters etc
- More advanced versions of the CPU, Memory and Disk checks
- This module is what is known as **active** checks, the Nagios XI server is actively performing the checking

[NSCA Agent] and [NSCA Commands]

NSClient 0.3.x is not compatible with the newer version of NSCA that comes with Nagios XI and hence this module cannot be used. You will need to upgrade to NSClient++ 0.4.x or newer to be able to use NSClient++ for **passive** checks.

[External Scripts]

This section will become available once the `CheckExternalScripts.dll` module has been loaded. The `CheckExternalScripts.dll` module is what allows custom scripts / plugins to be executed to provide more advanced monitoring of the Windows machine, it is utilized by the the `NSClientListener.dll` and `NSCAAgent.dll` modules.

The following KB article provided detailed examples on how to add your own scripts / plugins to this section:

<https://support.nagios.com/kb/article.php?id=528>

NSClient++ 0.3.x Summary

This is just a basic overview of the most common configuration options. As mentioned on the first page of this document, the developer of NSClient++ no longer supports the 0.3.x version and has since removed the documentation for the old version. You can try your luck on the “Internet Archive Way Back Machine” to view a cached copy of the documentation before it was taken down:

<http://web.archive.org/web/20141010171248/http://www.nsclient.org/nscp/wiki/doc/configuration/0.3.x>

Restarting The NSClient++ Service

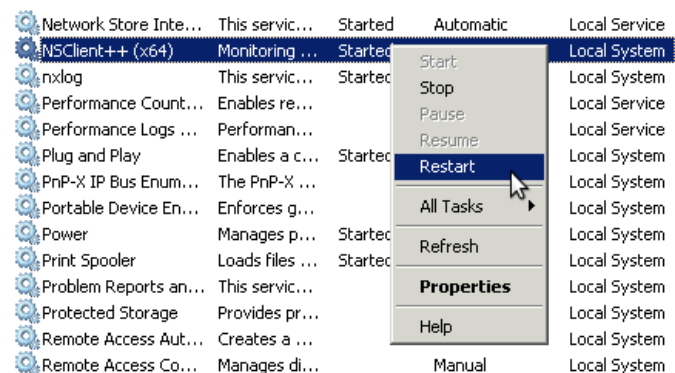
NSClient++ must now be restarted. In the Windows open the **Services** console under **Administrative Tools**.

If you cannot locate this, use `services.msc` to open the Services console.

Locate the **NSClient++** service.

Right click the **NSClient++** service and select **Restart**.

You can close the Services console as it's no longer required.



Troubleshooting

The most common problems, regarding the use of NSClient++ with Nagios XI are:

Misconfiguration in the `nsclient.ini` or `NSC.ini` file, which may include the following:

- Modules that you need to use are still commented out
- Password is not set or a wrong password is specified on the Nagios XI side
- Nagios XI server IP is not added to the `allowed hosts` line
- `allow arguments = 0` and you are trying to pass arguments (NRPE issue)
- The `CheckExternalScripts` module has not been loaded, but you are trying to run a script
- You forgot to restart the NSClient++ service after modifying the config file `nsclient.ini` or `NSC.ini`

Firewall Issues

Make sure all ports that need to be open can be accessed from the Nagios XI server:

- TCP port **12489** for `NSClientServer` (Nagios XI Server `check_nt` > Windows Machine)
- TCP port **5666** for `NRPEServer` (Nagios XI Server `check_nrpe` > Windows Machine)
- TCP port **5667** for `NSCAClient` (Windows Machine > Nagios XI Server)
- TCP port **80** or **443** for `NRDPCClient` (Windows Machine > Nagios XI Server)

You can test if a port is accessible by using the `nmap` command from the Nagios XI server. Establish a terminal session to your Nagios XI server as root and run the following command:

```
nmap <windows machine address> -p T:12489
```

The screenshots on the following page show the `nmap` command and output.

```
[root@xi-c6x-x64 libexec]# nmap 10.25.14.5 -p T:12489
Starting Nmap 6.47 ( http://nmap.org ) at 2016-11-04 17:10 AEDT
Nmap scan report for win7-04.box293.local (10.25.14.5)
Host is up (0.00029s latency).
PORT      STATE SERVICE
12489/tcp  open  unknown
MAC Address: 00:50:56:AB:E4:B5 (VMware)
Nmap done: 1 IP address (1 host up) scanned in 0.45 seconds
```

```
[root@xi-c6x-x64 libexec]# nmap 10.25.14.5 -p T:12489
Starting Nmap 6.47 ( http://nmap.org ) at 2016-11-04 17:10 AEDT
Nmap scan report for win7-04.box293.local (10.25.14.5)
Host is up (0.00036s latency).
PORT      STATE SERVICE
12489/tcp  filtered unknown
MAC Address: 00:50:56:AB:E4:B5 (VMware)
Nmap done: 1 IP address (1 host up) scanned in 0.24 seconds
```

If it shows an `open` state (left image) then NSClient++ is listening on that port and it is open.

If it shows a `filtered` or `closed` state (right image) then NSClient++ is NOT listening on that port or is being blocked by a firewall.

If you don't have `nmap` installed on the Nagios XI server, you can install it by running the following command:

```
yum install -y nmap
```

Check Log File

To troubleshoot issues, you can review the NSClient++ log file `nsclient.log`, the file is usually located in:

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

You may need to enable debugging in the `nsclient.ini` or `NSC.ini` file, refer to the earlier information in this document about the log section. Try scheduling an immediate check from your Nagios XI server against the client machine and then check the `nsclient.log` for errors.

Test Mode

Another way to find and solve problems is to run NSClient++ in "test" mode. This is done by stopping the service and starting it in "test" mode.

You can do this from the Windows Command Prompt (you may need to run it as administrator):

NSClient 0.3.x:

```
net stop nsclientpp
cd "C:\Program Files\NSClient++"
nsclient++ /test
```

To exit the "test" mode and start the service, run:

```
exit
net start nsclientpp
```

NSClient 0.4.x:

```
net stop nscp
cd "C:\Program Files\NSClient++"
nscp test
```

To exit the "test" mode and start the service, run:

```
exit
net start nscp
```

While you are in a "test" mode, you can run a check from the Nagios XI server against the client machine, and observe the output for errors.

Upgrade

Upgrading the NSClient++ to the latest version might resolve your issue in some circumstances.

Install From Scratch

Sometimes it's easier to uninstall NSClient++ and then re-install it to see if that fixes your problem. Make sure you delete the `C:\Program Files\NSClient++` directory to ensure you're performing a clean installation.

Additional Documentation

The following documentation will provide additional information about NSClient++ and how it can be used with Nagios XI:

- [Enabling The NRPE Listener In NSClient++ 0.3.x](#)
- [Enabling The NRPE Listener In NSClient++ 0.4.x](#)
- [Using NSClient++ For Passive Checks](#)
- [Integrating Autolt With Nagios XI](#)
- [Checking For Windows Updates With Nagios XI](#)
- [Monitoring Performance Counters in Nagios Using NSClient++](#)
- [Windows Memory Usage - Physical](#)
- [How to Restart A Windows Service With NRPE](#)
- [Using Scripts / Plugins With NSClient++](#)

Finishing Up

This completes the documentation on configuring the Windows agent NSClient++.

If you have additional questions or other support related questions, please visit us at our Nagios Support Forums:

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

The Nagios Support Knowledgebase is also a great support resource:

<https://support.nagios.com/kb>