# Nagios XI – How to Restart A Windows Service With NRPE

Nagios®

**The Industry Standard in IT Infrastructure Monitoring**

## Purpose

This document describes how to restart services in Windows using Nagios XI with NSClient++ via NRPE. This allows you to automate the process of starting, stopping and restarting Window Services.

## Target Audience

This document is intended for use by Nagios XI Administrators who want to automate starting, stopping, and restarting of Windows Services. A basic knowledge of NSClient++ and NRPE is recommended.

## Prerequisites

It is required you have NSClient++ installed, and NRPE configured on the Windows machine you intend to use for this task. NSClient++ must also be configured to allow NRPE checks from the Nagios server. This guide focuses on NSClient++ version 0.4.x and newer. Information on installing and configuring NSClient++ can be found in the following documents:

https://assets.nagios.com/downloads/nagiosxi/docs/Installing_The_XI_Windows_Agent.pdf
https://assets.nagios.com/downloads/nagiosxi/docs/Configuring_The_XI_Windows_Agent.pdf
https://assets.nagios.com/downloads/nagiosxi/docs/Enabling_the_NRPE_Listener_in_NSClient_0.4.x.pdf

## Create A Batch File To Restart The Service

On your windows machine open Notepad and paste in the following code:

```
@echo off
net stop %1
net start %1
@exit 0
```

Once completed, save it as a batch file called **`restart_service.bat`** in your NSClient++'s scripts directory:

```
C:\Program Files\NSClient++\scripts\
```

The `%1` argument is the name of the service, this will be received from an event handler which will be created later in this document.

## Configure NSClient++

Open `C:\Program Files\NSClient++\nsclient.ini` in Notepad and navigate to the `[/settings/external scripts/scripts]` section (if the section does not exist you will also need to add it).

Add the following line:

```
restart_service = scripts\restart_service.bat "$ARG1$"
```

```
[/settings/external scripts/scripts]
restart_service = scripts\restart_service.bat "$ARG1$"
```

Also, verify that `allow arguments = true` is configured. If this variable is not set to `0` or `false`, you will **not** be able to pass arguments to your scripts and the `restart_service.bat` script will not work. This must be configured in BOTH of these sections:

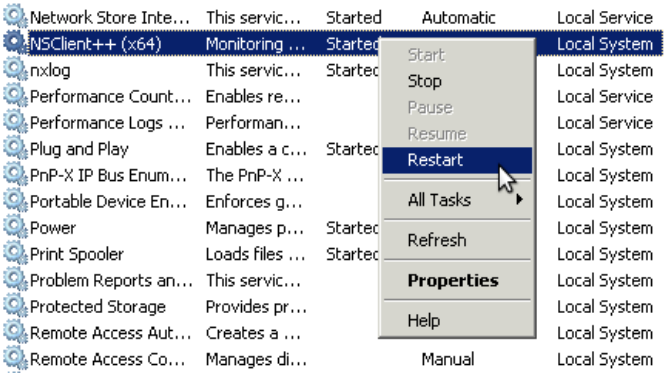
```
[/settings/NRPE/server]
[/settings/external scripts]
```

**Save** the `nsclient.ini` file.

NSClient++ must now be restarted to pick up the changes. 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.



## Test The Command From The Nagios Server

Now we will test from the Nagios XI server that the command you just added to NSClient++ is working. This example is going to restart the **spooler** service as it is unlikely to cause any issues. Establish a terminal session to your Nagios XI server and execute the following command:

```
/usr/local/nagios/libexec/check_nrpe -H 10.25.14.3 -p 5666 -c restart_service -a spooler
```

```
[root@xi-r5x-x86 libexec]# /usr/local/nagios/libexec/check_nrpe -H 10.25.14.3 -p 5666 -c restart_service -a spooler
The Print Spooler service is stopping.
The Print Spooler service was stopped successfully.

The Print Spooler service is starting.
The Print Spooler service was started successfully.
```

You can see from the screenshot that we received back the results from the restart_service command, it appears to be working.

## Create Event Handler Script

Next we need to create a script that will be used by Nagios XI for the event handler. The script will be called `restart_service.sh` and will be located in the `/usr/local/nagios/libexec/` directory on the Nagios XI server. Execute the following command:

```
vi /usr/local/nagios/libexec/restart_service.sh
```

*When using the vi editor, to make changes press **i** on the keyboard first to enter insert mode. Press **Esc** to exit insert mode.*

Paste the following into the terminal session:

```
#!/bin/sh
case "$1" in
	OK)
		;;
	WARNING)
		;;
	UNKNOWN)
		;;
	CRITICAL)
		/usr/local/nagios/libexec/check_nrpe -H "$2" -p 5666 -c restart_service -a "$3"
	;;
esac
exit 0
```

When you have finished, save the changes in vi by typing:

```
:wq
```

and press Enter.

Now execute the following commands to set the correction permissions:

```
chown apache:nagios /usr/local/nagios/libexec/restart_service.sh
chmod 775 /usr/local/nagios/libexec/restart_service.sh
```

You can now test the script works by executing the following command:

```
/usr/local/nagios/libexec/restart_service.sh CRITICAL 10.25.14.3 spooler
```

When the script is run, it receives three arguments which are referenced as `$1`, `$2`, `$3` in the script.

`$1` = The state of the service.

`$2` = The host address of the Linux server.

`$3` = The name of the service being restarted.

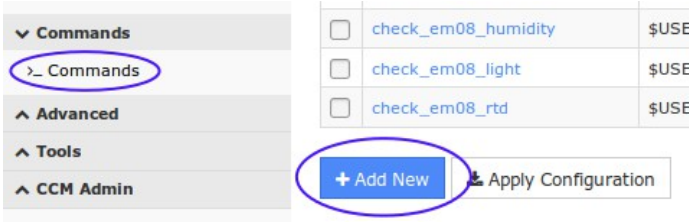
You can see from the script above that it's only when the service is in a CRITICAL state that the service_restart command will be executed.

## Create Event Handler

Now an event handler on the Nagios XI server will be created which will be used by your services.

Navigate to **Configure** > **Core Configuration Manager**

Select **Commands** from the list on the left, click the **>_ Commands** link and then click the **Add New** button.



You will need to populate the fields with the following values:

**Command**

```
Service Restart - Windows
```

**Command line**

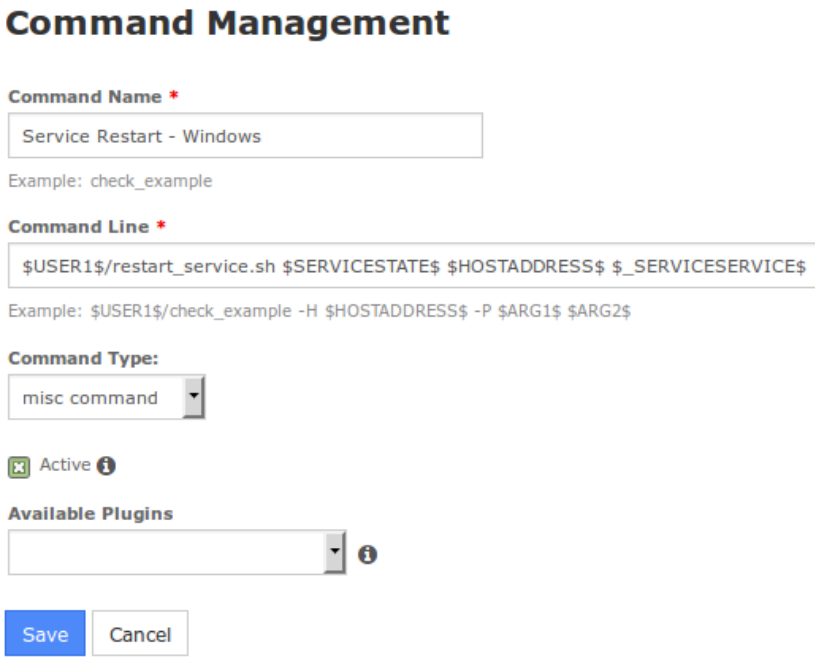
```
$USER1$/restart_service.sh $SERVICESTATE$ $HOSTADDRESS$ $_SERVICESERVICE$
```

**Command type**

misc command

Check the **Active** check box.

Click the **Save** button and then **Apply Configuration**.




## Adding a Service Check

Now we will need to create a Service using the Windows Server wizard. Navigate to **Configure** from the top menu of the Nagios XI web interface and select **Run a configuration wizard**.

Select **Windows Server** and click **Next**. Enter the IP address of the Windows Host you will be monitoring a service on and click **Next**.

On **Step 2** of the wizard you need to add **spooler** under the **Windows Service** field and **Print Spooler** in the **Display Name** field.

Finish the wizard to create the new service.

## Update Service With Event Handler

Now that the Nagios service is created we need to do two things:

- Select Event Handler
- Add the name of the service we want to restart as a custom variable to the service object. This is how the event handler knows what the name of the service is to restart.

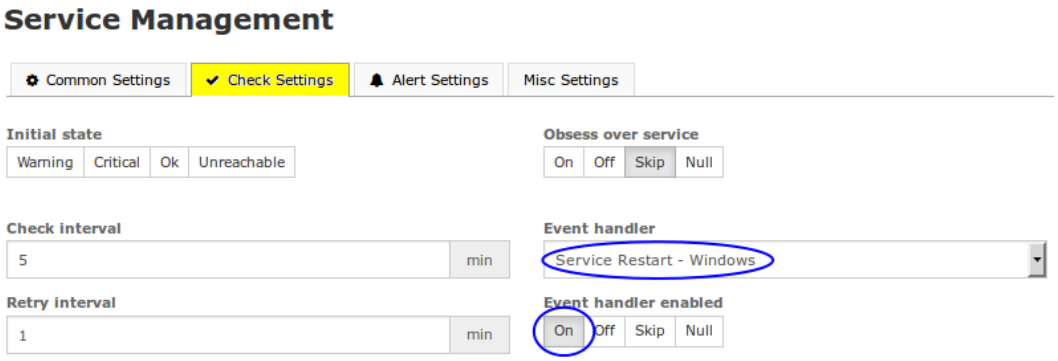
Navigate to **Configure** > **Core Configuration Manager** > **Monitoring** > **Services**.

Click the service **Print Spooler** to edit the service.

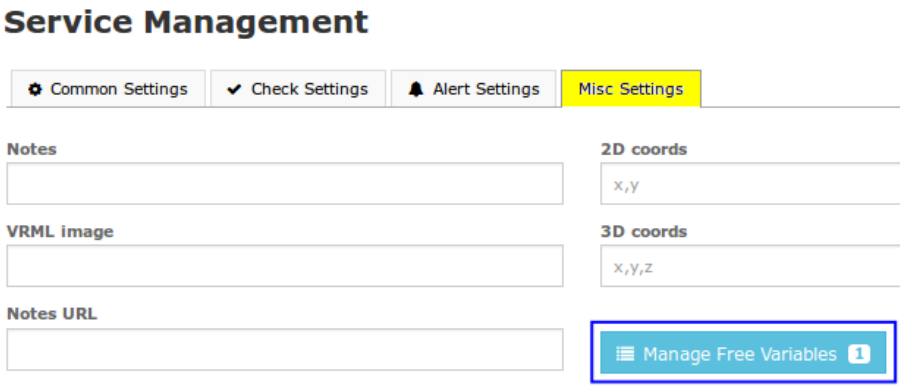
Click the **Check Settings** tab.

For the **Event handler** drop down list select the option **Service Restart - Windows**.

For **Event handler enabled** click **On**.




Click the **Misc Settings** tab and then click the Manage Free Variables button.



We will be adding a custom variable so that the event handler knows the name of the service to restart.

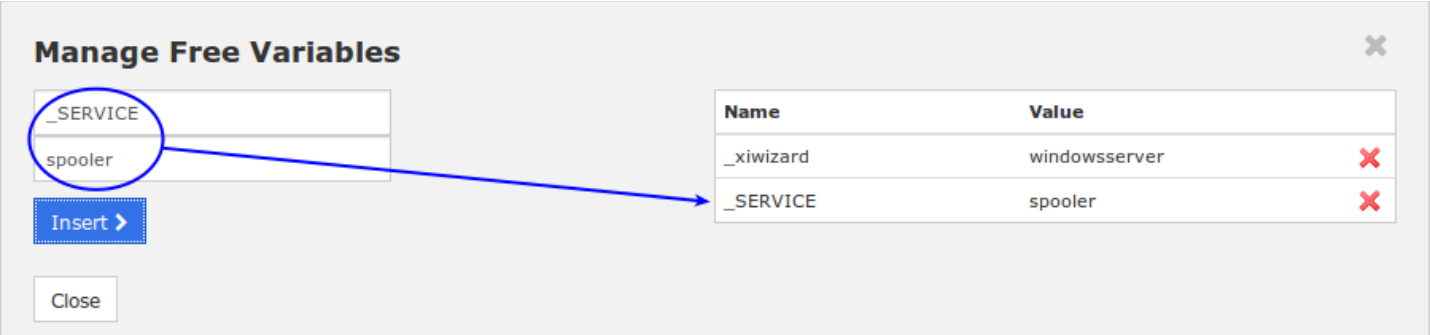
Name:

```
_SERVICE
```

Value:

```
spooler
```

Click **Insert** and the variable will be added to the list on the right.




Click the **Close** button and then click the **Save** button.

Click **Apply Configuration** for the changes to take affect.

In the event handler command you created, you can see the macro `$_SERVICESERVICE$` was used. This is how a service macro is referenced by the Nagios Core engine. More information on custom variables can be found here:

https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/customobjectvars.html

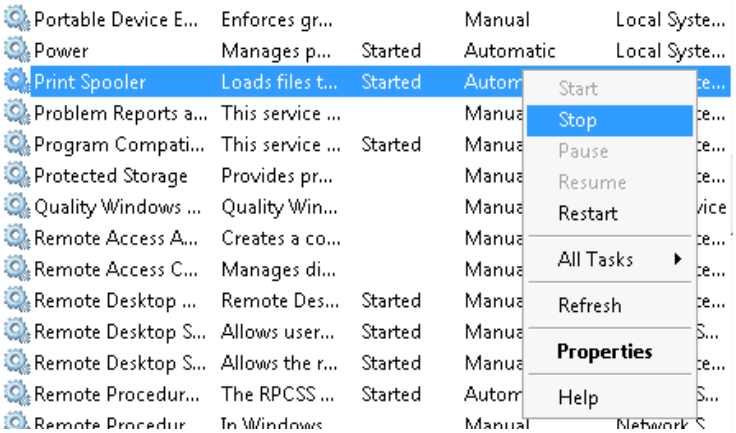
## Test

To test simply stop the Print Spooler service on the Windows machine.

Open the **Services** console under **Administrative Tools**.

**Right** click the **Print Spooler** service and select **Stop**.

Wait for the Nagios service to go to a critical state or force the next check.




Once the Nagios XI **Print Spooler** service is in a critical state the event handler will be executed and the Windows **Print Spooler** service will be restarted. The next time Nagios XI checks the **Print Spooler** service it will return to an OK state as the Windows **Print Spooler** service will now be running.

## Troubleshooting

If the event handler does not appear to be working as expected, check the `/usr/local/nagios/var/nagios.log` file for any errors, for example:

```
[1481763272] SERVICE ALERT: 10.25.14.3;Print Spooler;CRITICAL;SOFT;1;spooler: Stopped
[1481763272] wproc: SERVICE EVENTHANDLER job 7 from worker Core Worker 12627 is a non-check helper but exited with return code 13
[1481763272] wproc:   early_timeout=0; exited_ok=1; wait_status=3328; error_code=0;
[1481763272] wproc:   stderr line 01: execvp(/usr/local/nagios/libexec/restart_service.sh, …) failed. Errno is 13: Permission denied
```

In the log entries above you can see that the worker reported that it did not have permission to execute the `restart_service.sh` command.

## Finishing Up

This completes the guide on how to incorporate a NRPE check and event handler into Nagios XI which will automatically restart a Windows service if it goes into a critical state. If you have further questions please visit our support forums at:

https://support.nagios.com/forum