Installing NSClient++

This is a grooving process before it was all manual but slowly we are getting a more "automated" installation process so hopefully this will keep improving in the future as well and some of the steps might go away.

1. Installation

NSClient++ comes with an interactive installer (MSI) which should preferably be used. There is also a command line option for registering (and de-registering) the service for details refer to the manual installation guide. If you are using Windows NT4 there is some dependencies you need to manually install for details refer to the NT4 Dependency guide.

Details on how to do silent installs and automated silent installs can be found at the Installation Guide page.

Thus to install the Client you simply click the MSI package (for your platform) and follow the wizard through. **BUT** and this is a big but after you have installed it it still needs to be configure (which is done with your favorite text editor).

2. Configuration

Before you start NSClient++ you need to configure it by editing the configuration file (NSC.ini). The configuration file is a simple text file and is explained in detail under Configuration.

The configuration file (NSC.ini) **NEEDS** to be configured as for security reasons all plug-ins are disabled by default. The reason for this is so no one will accidentally install this and get potential security issues, I believe that things should be "off" by default. Also notice that by default allowed_hosts are 127.0.0.1 so you need to modify this as well.

3. System tray

If you plan to use the SystemTray module (that shows a system tray icon on the desktop you need to install the SystemTray module as well as NSClient++ on "old" versions of windows (XP and blow) on modern version of windows (XP and above) you can used the new experimental shared session support. For details on this see the System tray installation guide.

4. Testing and Debugging

After you have installed NSClient++ you need to start it which is done which can be done in several ways as it is a normal service (so either fire up a command line and use the net start/stop command of you can use the computer manager services node.

When you are starting yout and/or configuring your client you can use the "debug" mode which will be very helpfull as you will see the debug log in "real time" whrn you play around with it. To start NSClient++ in test/debug mode use the following command (you can also use the icon on the start menu):

```
NSClient++ /test
```

5. Windows Firewall

I have yet to foigure this one out but hopefully someone can help me write this! I shall for the next version try to make an automated exception thingy for the windows firewall.

6. External Firewall (optional)

Firewall configuration should be pretty straight forward:

- If you use NRPEListener (check_nrpe) you need the NRPE port open (usually 5666) from the nagios server towards the client.
- If you use the NSClientListener (check_nt) you need the (modified) NSClient port open (usually 12489) from the nagios server towards the client.
- If you use the NSCA Module (passive checks) you need the NSCA port open from the client towards the nagios server. client:* -> nagios:5667
- If you use the NRPEClient module to check any remote systems (use NSClient++ as a proxy) you need to have NRPE port (usually 5666) open from NSClient++ (the proxy) to the remote-client in addition to the method you use to submit the results to the server. nsclient-proxy:* -> remote-client:5666

Protocol	Source	Source port	Destination	Destination port	Comment
NRPE	**nagios**	<all>	Client	5666	The nagios server initiates a call to the client on port 5666
NSClient	**nagios**	<all>	Client	12489	The nagios server initiates a call to the client on port 12489
NSCA	client	<all>	**nagios**	5667	The client initiates a call to the nagios server on port 5667
NRPE-proxy	client	<all>	remote-client	5666	The client initiates a call to the remote client on port 5666

- **nagios** Is the ip/host of the main nagios server
- client is the windows computer where you have installed NSClient++
- remote-client is the "other" client you want to check from NSClient++ (using NSClient++ as a proxy)

All these ports can be changed so check your nsc.ini.