

How To Create Nagios Plugins With Bash On Ubuntu 12.10

Authored by: **ASPHostServer Administrator** [asphostserver@gmail.com]

Saved From: <http://faq.asphosthelpdesk.com/article.php?id=278>

This time, we will expand on this idea and create Nagios plugins using Bash.

Step 1 - Install NRPE on client

```
apt-get install -y nagios-nrpe-server
useradd nrpe && update-rc.d nagios-nrpe-server defaults
```

Step 2 - Create your Bash Script It would be a good idea to keep your plugins in same directory as other Nagios plugins (`/usr/lib/nagios/plugins/` for example). For our example, we will create a script that checks current disk usage by calling "df" from shell, and throw an alert if it is over 85% used:

```
#!/bin/bash
used_space=`df -h / | grep -v Filesystem | awk '{print }' | sed 's/%//g'`
case $used_space in
[1-84]*)
echo "OK - $used_space% of disk space used."
exit 0
;;
[85]*)
echo "WARNING - $used_space% of disk space used."
exit 1
;;
[86-100]*)
echo "CRITICAL - $used_space% of disk space used."
exit 2
;;
*)
echo "UNKNOWN - $used_space% of disk space used."
exit 3
;;
esac
```

Exit Code	Status
	OK
1	WARNING
2	CRITICAL
3	UNKNOWN

We will save this script in `/usr/lib/nagios/plugins/usedspace.sh` and make it executable:

```
chmod +x /usr/lib/nagios/plugins/usedspace.sh
```

The entire Nagios NRPE plugin boils down to using exit codes to trigger alerts.

You introduce your level of logic to the script, and if you want to trigger an alert (whether it is OK, WARNING, CRITICAL, or UNKNOWN) - you specify an exit code.

Refer to the following Nagios Exit Codes:

Nagios Exit Codes Step 3 - Add Your Script to NRPE configuration on client host Delete original `/etc/nagios/nrpe.cfg` and add the following lines to it:

```
log_facility=daemon
pid_file=/var/run/nagios/nrpe.pid
server_port=5666
nrpe_user=nrpe
nrpe_group=nrpe
allowed_hosts=198.211.117.251
dont_blame_nrpe=1
debug=0
command_timeout=60
connection_timeout=300
include_dir=/etc/nagios/nrpe.d/
command[usedspace_bash]=/usr/lib/nagios/plugins/usedspace.sh
```

Where 198.211.117.251 is our monitoring server from previous articles. Change these to your own values.

Make sure to restart Nagios NRPE service:

```
service nagios-nrpe-server restart
```

Step 4 - Add Your New Command to Nagios Checks on Nagios Monitoring Server Define new command in `/etc/nagios/objects/commands.cfg`

```
define command{
    command_name    usedspace_bash
    command_line    $USER1$/check_nrpe -H $HOSTADDRESS$ -c usedspace_bash
}
```

As you can see, it uses NRPE to make TCP connections to port 5666 and run command 'usedspace_bash', which we defined in `/etc/nagios/nrpe.cfg` on that remote host.

Add this check to your Nagios configuration file for client.

For our example, we will monitor a server called MyUbuntu and edit `/etc/nagios/servers/MyUbuntu.cfg`

```
define service {
    use                generic-service
    host_name          MyUbuntu
    service_description Custom Disk Checker In Bash
    check_command      usedspace_bash
}
```

Restart Nagios:

```
service nagios restart
```

Verify that the new check is working. And you are all done!