

Nagios Checks in Depth

Systems Administration

School of Information Technology
Otago Polytechnic
Dunedin, New Zealand

Three kinds of Nagios Checks

- 1 Local services
- 2 Network exposed services
- 3 Remote services

Local Services

- Services running on the same system that runs Nagios
- A good way to explore the mechanics of plugins and checks

in /etc/nagios3/conf.d/localhost_nagios2.cfg

```
define service{
    use                generic-service
    host_name          localhost
    service_description Disk Space
    check_command       check_all_disks!20%!10%
}
```

in /etc/nagios-plugins/config/disk.cfg

```
define command{  
    command_name    check_all_disks  
    command_line    /usr/lib/nagios/plugins/check_disk ...  
                   -w '$ARG1$' -c '$ARG2$' -e  
}
```

check_disk

This is just a simple program written in C. We can call it manually.

```
root@app:~# /usr/lib/nagios/plugins/check_disk -w 20% -c 10%
DISK OK - free space: / 5887 MB (80% inode=90%);
/lib/init/rw 122 MB (100% inode=99%);
/dev 117 MB (99% inode=98%);
/dev/shm 122 MB (100% inode=99%);|
/=1405MB;6146;6914;0;7683
/lib/init/rw=0MB;97;109;0;122
/dev=0MB;93;105;0;117 /dev/shm=0MB;97;109;0;122
```

Network Exposed Services

- Very similar to local services
- Nothing extra needs to be installed on the monitored systems
- Just connect to the service over the network and see if it works

For example, lets check MySQL.

in /etc/nagios3/conf.d/services_nagios2.cfg

The MySQL service is not defined, so add:

```
#check that mysql services are running
define service {
    hostgroup_name      mysql-servers
    service_description MySQL
    check_command        check_mysql_cmdlinecred!$USER3$!$USER4$
    use                  generic-service
    notification_interval 0
}
```


in /etc/nagios3/resources.cfg

This file contains items we need, but that need to be handled carefully, like usernames and passwords. Add:

```
# Store some usernames and passwords (hidden from the CGIs)
# MySQL username and password
$USER3$=root
$USER4$=foo

}
```

in /etc/nagios-plugins/config/mysql.cfg

```
# 'check_mysql_cmdlinecred' command definition
define command{
    command_name    check_mysql_cmdlinecred
    command_line    /usr/lib/nagios/plugins/check_mysql
                    -H '$HOSTADDRESS$' -u '$ARG1$' -p '$ARG2$'
}
```

We still need to do three things

- 1 Define a host
- 2 Define a hostgroup, mysql-servers
- 3 Put our new host in the hostgroup

in /etc/nagios3/conf.d/hostgroups_nagios2.cfg

```
# A list of your mysql servers
define hostgroup {
    hostgroup_name  mysql-servers
        alias      MySQL servers
        members     db
    }
```

in /etc/nagios3/conf.d/db_host.cfg

```
# Database server host definitions

define host{
    use                generic-host
    host_name          db
    alias              db
    address            192.168.2.102
}
```

Ready to monitor MySQL

Now we can restart Nagios and we'll be monitoring our MySQL server.

Remote Services

- Sometimes we want to monitor remote services that are not exposed on a network
- There are a few ways to handle this, each with its pros and cons
- We'll consider one way, using *NRPE*

The approach

- We install the NRPE daemon on the remote host we want to monitor
- We also install the desired monitoring plugins on the remote host
- On the Nagios server, we use the NRPE plugin to send the check request to the remote host
- On the remote host, the NRPE daemon runs the check and reports the results back to the Nagios server

Install the NRPE daemon

On the remote server you want to monitor

- `sudo apt-get install nagios-nrpe-server`
- Edit `/etc/nagios/nrpe.cfg`
- Add 192.168.2.103 to the `allowed_hosts` values
- Add the following command to the `commands` section
`command[check_disk]=/usr/lib/nagios/plugins/check_disk -w 20%
-c 10%`
- Restart the NRPE server

On the Nagios server

Modify `/etc/nagios3/conf.d/db_host.cfg` by adding

```
define service{
    use                generic-service
    host_name          db
    service_description Disk Space
    check_command       check_nrpe_1arg!check_disk
}
```

Monitoring Windows Hosts

This done with a tool that is similar to NRPE

- On the Windows host, we install NSClient++
- On the Nagios server, we use the `check_nt` plugin to communicate with the Windows host