

Contents

- 1 Nagios is a Network Monitoring Service :: Setup and Install
- 2 Configuration of Nagios
 - ◆ 2.1 Overview of Nagios Config Files and Plugins
 - ◆ 2.2 0. Additional Info Available
 - ◆ 2.3 1. Config all unique hosts
 - ◇ 2.3.1 Disable Checking of a Host
 - ◆ 2.4 2. Config Nagios hostgroups
 - ◆ 2.5 3. Config Nagios Contacts
 - ◆ 2.6 4. Config Nagios Contactgroups
 - ◆ 2.7 5. Config Nagios Services
 - ◆ 2.8 6. Extra Custom Plugin Configs
- 3 Send Nagios Notifications via SMS Text Messages
- 4 References & Additional Info
- 5 Monitor HTML via a Proxy

Nagios is a Network Monitoring Service :: Setup and Install

It can monitor several services on several hosts and notify by email etc. a certain group depending on the levels of measurement. To keep it simple:

```
apt-get install nagios-text
```

On the install process, a password for the default admin user is required.

```
nagiosadmin
password_chosen_at_install (This is for the Web Interface)
```

additional users: /etc/nagios/htpasswd.users (add via apache htpasswd)

There is a ton of configuring to be done. First off - apache2 site-enabled.

```
ln -s /etc/nagios/apache.conf /etc/apache2/sites-enabled/nagios
(restart apache)
```

This will get the basics done at <http://localhost/nagios>. You will be able to login. The Default Gateway should get added in by default and will be monitored ok. Copy the settings in /etc/nagios and put in another host etc...

Great Explanation at: <http://www.debian-administration.org/articles/299>

Configuration of Nagios

There is quite a bit of configuration required for Nagios. If the following steps are carried out in order, things should be a lot easier. Although by default the "Default Gateway" (gw) is added in with its own group etc. it was put into a new hostgroup with updated contact details.

Overview of Nagios Config Files and Plugins

The main nagios config files are kept in: /etc/nagios/ The plugin config files are kept in: /etc/nagios-plugins/config/ The executable plugins are kept in: /usr/lib/nagios/plugins/

0. Additional Info Available

Please read http://nagios.sourceforge.net/docs/2_0/xodtemplate.html#host for all details relating to the options/files below and their template. E.g. the following host config options are explained there: d,u,r. d=down. u=unreachable. r=recovered (note: there are more options available). Extended example configs are located at: /usr/share/doc/nagios-text/examples/template-object/

1. Config all unique hosts

Note: Only specify different physical servers (ip's). Multiple http websites can be monitored on 1 host. vi /etc/nagios/hosts.cfg

```
define host{
    name                generic-host      ; The name of this host template....
    notifications_enabled 1                ; Host notifications are enabled
    event_handler_enabled 1                ; Host event handler is disabled
    flap_detection_enabled 0               ; Flap detection is disabled. Flap = prevents against
    process_perf_data     1                ; Process performance data
    retain_status_information 1            ; Retain status information across program restarts.
    retain_nonstatus_information 1         ; Retain non-status information across program restarts.
    register              0                ; DONT REGISTER THIS DEFINITION - ITS NOT A REAL HOST
}

# Default gateway host definition
define host{
    use                generic-host      ; Name of host template to use
    host_name          gateway
    alias              Default Gateway
    address            ip.address.or.domain.com.name
    check_command      check-host-alive
    max_check_attempts 20
    notification_interval 60
    notification_period 24x7
    notification_options d,u,r
}

define host{
    use                generic-host      ; Name of host template to use
    host_name          domain1.com
    alias              Domain 1
    address            ip.or.host.name
    check_command      check-host-alive
    max_check_attempts 20
    notification_interval 120
    notification_period 24x7
    notification_options d,u,r
}

define host{
    use                generic-host      ; Name of host template to use
```

Netsaint / Nagios

```
host_name          domain2.com
alias              Domain 2
address            ip.address.or.host.name
check_command      check-host-alive
max_check_attempts 20
notification_interval 120
notification_period 24x7
notification_options d,u,r
}

define host{
    use                generic-host          ; Name of host template to use
    host_name          www.google.com
    alias              Google Webserver
    address            www.google.com
    check_command      check-host-alive
    max_check_attempts 20
    notification_interval 120
    notification_period 24x7
    notification_options d,u,r
}
```

Disable Checking of a Host

I have been having problems with 1 host in particular, where nagios gets tied up checking TTL and does not wait between TTL checks. The errors were:

```
[06-24-2007 11:10:13] HOST ALERT: host.com;DOWN;SOFT;19;CRITICAL - Time to live exceeded (82.195.144
[06-24-2007 11:10:13] HOST ALERT: host.com;DOWN;SOFT;18;CRITICAL - Time to live exceeded (82.195.144
[06-24-2007 11:10:13] HOST ALERT: host.com;DOWN;SOFT;17;CRITICAL - Time to live exceeded (82.195.144
[06-24-2007 11:10:13] HOST ALERT: host.com;DOWN;SOFT;16;CRITICAL - Time to live exceeded (82.195.144
#and so on for 20 checks with no wait
```

The same error has been discussed and described further here:

<http://readlist.com/lists/lists.sourceforge.net/nagios-users/0/2181.html> Instead of putting in some code to get nagios waiting between TTL checks, I simply chose to disable host checking, and to check just the service on that server instead. To disable checking of a host, add the following to the define host{ } code (as above):

```
define host{
    use                generic-host          ; Name of host template to use
    host_name          www.google.com
    alias              Google Webserver
    address            www.google.com
    check_command      check-host-alive
    max_check_attempts 20
    checks_enabled     0
    notification_interval 120
    notification_period 24x7
    notification_options d,u,r
}
```

2. Config Nagios hostgroups

Hostgroups quite simply group together all the hosts in hosts.cfg. They are mainly used to order and group services and hosts together. I created separate hostgroups for various server clusters. I.e. 1 hostgroup for my own

1. Config all unique hosts

Netsaint / Nagios

server cluster, and a second for my computer society servers, and a third for Commerical Hosting web servers.

```
vi /etc/nagios/hostgroups.cfg
```

```
define hostgroup{
    hostgroup_name my_cluster
    alias          My Server Cluster
    contact_groups root-my_cluster
    members       gateway, domain1.com, domain2.com
}

define hostgroup{
    hostgroup_name other-webservers
    alias          Other Commercial Web Servers
    contact_groups select-users-my_cluster
    members       www.google.com
}
```

3. Config Nagios Contacts

Note: As with hosts, the contacts config takes in specific names of people and their contact information. Various contacts are then grouped together in step 4. For this config, I am going to have 2 main contacts. 1 is going to be the root administrator and the second is going to be a general user (for receiving information on the non essential other-webservers). Again, look at http://nagios.sourceforge.net/docs/2_0/xodtemplate.html#contact for specifics on notification options.

```
vi /etc/nagios/contacts.cfg
```

```
define contact{
    contact_name      root
    alias             Root Administrator
    service_notification_period 24x7
    host_notification_period 24x7
    service_notification_options w,u,c,r
    host_notification_options d,u,r
    service_notification_commands notify-by-email
    host_notification_commands host-notify-by-email
    email            root@domain.com
}

define contact{
    contact_name      sburke
    alias             A Standard/Typical User
    service_notification_period 24x7
    host_notification_period 24x7
    service_notification_options w,u,c,r
    host_notification_options d,u,r
    service_notification_commands notify-by-email
    host_notification_commands host-notify-by-email
    email            username@domain.com
}
```

4. Config Nagios Contactgroups

Again, all the various contacts as outlined in step 3 needs to be grouped together. The `hostgroups.cfg` and `services.cfg` send alert notifications to "contactgroups" and not individual contacts. Although all these separate configs seem to be very awkward, they ensure that users and hosts and services can be added easily.

```
vi /etc/nagios/contactgroups.cfg

define contactgroup{
    contactgroup_name    root-my_cluster
    alias                Root Admins on My Cluster
    members              root
}

define contactgroup{
    contactgroup_name    select-users-my_cluster
    alias                Users on Burkesys
    members              sburke
}
```

Note: "root-my_cluster", "root", "select-users-my_cluster" and "sburke" were selected from Steps 2 and 3.

5. Config Nagios Services

This is the main and final configuration file (typically). All information in the previous 4 steps must be used and matched up correctly with the configs and information in this step, otherwise nagios will complain and give a helpful debug.

```
vi /etc/nagios/services.cfg

# Generic service definition template
define service{
    ; The 'name' of this service template, referenced in other service definitions
    name                generic-service
    active_checks_enabled    1        ; Active service checks are enabled
    passive_checks_enabled    1        ; Passive service checks are enabled/disabled
    parallelize_check        1        ; Active service checks should be parallelized
                                ; (disabling this can lead to major performance probl
    obsess_over_service        1        ; We should obsess over this service (if necessary)
    check_freshness            0        ; Default is to NOT check service 'freshness'
    notifications_enabled        1        ; Service notifications are disabled
    event_handler_enabled        1        ; Service event handler is disabled
    flap_detection_enabled        0        ; Flap detection is disabled
    process_perf_data            1        ; Process performance data
    retain_status_information        1        ; Retain status information across program restarts
    retain_nonstatus_information        1        ; Retain non-status information across program restarts
    register                0        ; DONT REGISTER THIS DEFINITION - ITS NOT A REAL SERV
}

# Service definition
define service{
    use                generic-service        ; Name of service template to use
    host_name            domain1.com, domain2.com
    service_description        PING
    is_volatile            0
    check_period            24x7
    max_check_attempts        3
```

Netsaint / Nagios

```
normal_check_interval      5
retry_check_interval       1
contact_groups             root-my_cluster
notifications_enabled      1
notification_interval     120
notification_period        24x7
notification_options       w,u,c,r
check_command              check_ping!100.0,20%!500.0,60%
;check_ping syntax: !warning if exceeds 100ms,warning if exceeds 20% packet loss!critical if
}

define service{
    use                generic-service
    host_name          domain1.com
    service_description HTTP
    is_volatile        0
    check_period       24x7
    max_check_attempts 3
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     root-my_cluster
    notification_interval 120
    notification_period 24x7
    notification_options c,r
    check_command      check_http
}

define service{
    use                generic-service
    host_name          domain1.com
    service_description HTTP-vhost_name
    is_volatile        0
    check_period       24x7
    max_check_attempts 3
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     root-my_cluster
    notification_interval 120
    notification_period 24x7
    notification_options c,r
    check_command      check_http_url!http://vhost.domain1.com/path/to/application/p
}

define service{
    use                generic-service
    host_name          domain2.com
    service_description DNS
    is_volatile        0
    check_period       24x7
    max_check_attempts 3
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     root-my_cluster
    notification_interval 120
    notification_period 24x7
    notification_options c,r
    check_command      check_dns
}

define service{
    use                generic-service
    host_name          domain2.com
```

Netsaint / Nagios

```
service_description MySQL
is_volatile 0
check_period 24x7
max_check_attempts 3
normal_check_interval 5
retry_check_interval 1
contact_groups root-my_cluster
notification_interval 120
notification_period 24x7
notification_options c,r
check_command check_mysql_cmdlinecred!mysqluser!mysqlpassword
}

define service{
    use generic-service
    host_name domain2.com
    service_description SMTP
    is_volatile 0
    check_period 24x7
    max_check_attempts 3
    normal_check_interval 5
    retry_check_interval 1
    contact_groups root-my_cluster
    notification_interval 120
    notification_period 24x7
    notification_options c,r
    check_command check_smtp
}
#####
define service{
    use generic-service
    host_name www.google.com
    service_description PING
    is_volatile 0
    check_period 24x7
    max_check_attempts 3
    normal_check_interval 5
    retry_check_interval 1
    contact_groups select-users-my_cluster
    notification_interval 120
    notification_period 24x7
    notification_options c,r
    check_command check_ping!100.0,20%!500.0,60%
}

define service{
    use generic-service
    host_name www.google.com
    service_description HTTP
    is_volatile 0
    check_period 24x7
    max_check_attempts 3
    normal_check_interval 5
    retry_check_interval 1
    contact_groups select-users-my_cluster
    notification_interval 120
    notification_period 24x7
    notification_options c,r
    check_command check_http
}
```

The services.cfg can get quite long indeed! Services can be grouped together in servicegroups.cfg, however I didnt bother with this step. It provides a better overview using the Web Front end when there are a large number of services.

6. Extra Custom Plugin Configs

In the services.cfg, there is an "check_http_url" config added in. Currently nagios would give an error at this step. That is because "check_http_url" is a special config to monitor a vhost.domain1.com and prevents us from having to make a host for a virtual website to monitor.

```
vi /etc/nagios-plugins/config/http.cfg

# 'check_http3' command definition
define command{
    command_name      check_http_url
    command_line      /usr/lib/nagios/plugins/check_http -I $HOSTADDRESS$ -u $ARG1$
}
```

In order to see what options are available and the command line switches etc. do the following:

```
/usr/lib/nagios/plugins/check_http --help
```

There are several options for all of the plugins within /usr/lib/nagios/plugins/ to monitor various specific levels of performance.

Another config is /etc/nagios/escalations.cfg however at the moment I feel it works ok without this step. I will revisit it at a later stage.

Send Nagios Notifications via SMS Text Messages

Although a simple config could be made for nagios to send sms's via vodasms (o2sms), I chose to do the sms handling at email delivery time using procmail. Read more here:

[Vodasms#Forward Emails via SMS Text Message](#)

References & Additional Info

Vhost & Website Monitoring: <http://theories.darwinsys.com/2007/04/05/1175779980000.html>

Monitoring tomcat website: http://nagios.org/faqs/viewfaq.php?faq_id=310

<http://www.kernel-panic.it/openbsd/nagios/nagios3.html>

Main Nagios Templates and Docs: http://nagios.sourceforge.net/docs/2_0/xodtemplate.html

General: <http://www.onlamp.com/pub/a/onlamp/2002/09/26/nagios.html?page=1>

General and Good: <http://www.debian-administration.org/articles/299>

General with some mistakes: <http://servers.linux.com/servers/04/09/14/2317206.shtml>

MySQL info and Nagios: <http://www.gatorlug.org/files/GatorLUG.ppt>

Monitor HTML via a Proxy

If nagios is running on a server which its firewall blocks outgoing http(s) requests, then you will have to use a proxy (if available) to check http on a remote host/server. Here is the configs and tweaks required: vi /etc/nagios-plugins/config/http.cfg

```
# 'check_http_via_proxy
define command{
    command_name    check_http_via_proxy
    command_line    /usr/lib/nagios/plugins/check_http -H $ARG1$ -p $ARG2$ -u $ARG3$ -e 'HTTP/1.0
}

```

```
vi /etc/nagios/services.cfg
# edit the check_command for the particular service you require to:
host_name          externalserver.com
check_command      check_http_via_proxy!proxy.internalserver.com!3128!http://externalserver.com
# note - sometimes the squid proxy would only serve a cached page. To get around this, the check_command
check_command      check_http_via_proxy!proxy.internalserver.com!3128!http://externalserver.com

```

Hopefully that should work ok.