



Grid Monitoring with Nagios

Aries Hung, Joanna Huang, Felix Lee, Min Tsai

ASGC

WLCG T2 Asia Workshop

TIFR, Dec 2, 2006



Agenda

- Nagios Overview
- Nagios Installation and Configuration
- Plugin Development
- ASGC Plugins
- SMS System



Grid Monitoring

- Large scale resources in Grid environments
 - Large number of hosts, services and network resources
- Automatic and continuous monitoring in demand
 - Help sites to monitor Grid resources more effectively and efficiently
 - Not just to know when service breaks and fix it immediately
 - Learn more to increase performance of the grid services
 - What breaks the most?
 - What the usage patterns?
 - Where the bottlenecks lie?
 - What resources are required?
 - What common problems and specific issues?



Nagios Overview and Features I

- Nagios is an open source monitoring framework
 - Monitor:
 - **Network services** (SMTP, POP3, HTTP, NNTP, PING, etc.)
 - **Host resources** (load, disk, memory, running procs, log files, etc.)
 - Monitoring results and reports accessible via web interface
 - **Simple plugin design**: easy to extend
 - **Notification** of events (via email, pager, or other user-defined methods)
 - **Event handlers** that run in response to events for proactive problem resolution



Nagios Overview and Features II

- External command interface that allows on-the-fly modifications to be made to the monitoring and notification behavior through the use of the web interface
- Scheduled downtime for suppressing host and service notifications during periods of planned outages
- Ability to acknowledge problems via the web interface



Nagios Requirements

- Nagios runs on Unix and its variants
- Nagios optionally requires a Web server to be installed (for the Web interface)



Nagios: Server Installation (1/3)

- Acquire the following latest packages from <http://www.nagios.org/download/>
 - nagios-2.6.tar.gz
 - nagios-plugins-1.4.5.tar.gz

- Make a directory for placing the packages that you download

```
root@nagios ~]# mkdir /root/nagiosinstall
```

- Create the necessary directories, permissions and user accounts to run Nagios

```
root@nagios ~]# useradd nagios
```

```
root@nagios ~]# mkdir /usr/local/nagios
```

```
root@nagios ~]# mkdir /usr/local/nagios/libexec
```

```
root@nagios ~]# chown -R nagios:nagios /usr/local/nagios
```

```
root@nagios ~]# groupadd nagcmd
```

```
root@nagios ~]# usermod -G nagcmd apache
```

```
root@nagios ~]# usermod -G nagcmd nagios
```

```
root@nagios ~]# chgrp -R nagcmd /usr/local/nagios/var/rw
```

- Install the necessary dependencies using yum

```
root@nagios ~]# yum install gd-devel
```



Nagios: Server Installation (2/3)

- Go into the nagiosinstall directory and extract the Nagios tarball that you downloaded

```
root@nagios ~]# cd /root/nagiosinstall
root@nagios nagiosinstall]# tar -xzvf nagios-2.6.tar.gz
```

- Go into the newly created nagios-2.6 directory to compile and install nagios

```
root@nagios nagiosinstall]# cd nagios-2.6
root@nagios nagios-2.6]# ./configure --prefix=/usr/local/nagios --with-
    cgiurl=/nagios/cgi-bin --with-htmurl=/nagios --with-nagios-
    user=nagios --with-nagios-group=nagios --with-command-group=nagcmd
root@nagios nagios-2.6]# make all
root@nagios nagios-2.6]# make install
root@nagios nagios-2.6]# make install-init
root@nagios nagios-2.6]# make install-commandmode
root@nagios nagios-2.6]# make install-config
```




Nagios: Server Installation (3/3)

- Install the standard Nagios Plug-Ins

```
root@nagios nagios-2.6]# cd /root/nagiosinstall/
root@nagios nagiosinstall]# tar -xzvf nagios-plugins-1.4.5.tar.gz
root@nagios nagiosinstall]# cd /nagios-plugins-1.4.5
root@nagios nagios-plugins-1.4.5]# ./configure--prefix=/usr/local/nagios
--with-nagios-user=nagios --with-nagios-group=nagios --with-
cgiurl=nagios/cgi-bin
root@nagios nagios-plugins-1.4.5]# make
root@nagios nagios-plugins-1.4.5]# make install
```



Nagios: Server Configuration (1/5)

- Configure Apache for the Nagios Monitoring web site
 - Add 'Include /usr/local/nagios/etc/nagios-server.conf' to the bottom of the /etc/httpd/conf/httpd.conf file
- Create a file named /usr/local/nagios/etc/nagios-server.conf and insert the following into that file:

```
ScriptAlias /nagios/cgi-bin "/usr/local/nagios/sbin/"
<Directory "/usr/local/nagios/sbin/">
    Options ExecCGI
    AllowOverride None
    Order allow,deny
    Allow from all
    AuthName "Nagios Access"
    AuthType Basic
    AuthUserFile /usr/local/nagios/etc/htpasswd.users
    Require valid-user
</Directory>
```

```
Alias /nagios "/usr/local/nagios/share/"
<Directory "/usr/local/nagios/share/">
    Options None
    AllowOverride None
    Order allow,deny
    Allow from all
    AuthName "Nagios Access"
    AuthType Basic
    AuthUserFile /usr/local/nagios/etc/htpasswd.users
    Require valid-user
</Directory>
```



Nagios: Server Configuration (2/5)

- Create a file named `/usr/local/nagios/sbin/.htaccess` and insert the following into that file:

```
AuthName "Nagios Access"  
AuthType Basic  
AuthUserFile /usr/local/nagios/etc/htpasswd.users  
require valid-user
```

- Create a 'nagiosadmin' user account that will be used when prompted for authentication when accessing the Nagios web page

```
root@nagios nagios-plugins-1.4.5]# htpasswd -c  
/usr/local/nagios/etc/htpasswd.users nagiosadmin
```

- Setup the `cgi.cfg` file by doing the following

```
root@nagios nagios-plugins-1.4.5]# cd /usr/local/nagios/etc  
root@nagios etc]# mv cgi.cfg-sample cgi.cfg
```



Nagios: Server Configuration (3/5)

- Open the `cgi.cfg` file and un-comment the following:

```
authorized_for_system_information=nagiosadmin
authorized_for_configuration_information=nagiosadmin
authorized_for_system_commands=nagiosadmin
authorized_for_all_services=nagiosadmin
authorized_for_all_hosts=nagiosadmin
authorized_for_all_service_commands=nagiosadmin
authorized_for_all_host_commands=nagiosadmin
```

- Make the sample config files be your actual configuration files for Nagios

```
root@nagios etc]# mv checkcommand.cfg-sample checkcommands.cfg
root@nagios etc]# mv minimal.cfg-sample minimal.cfg
root@nagios etc]# mv misccommands.cfg-sample misccommands.cfg
root@nagios etc]# mv nagios.cfg-sample nagios.cfg
root@nagios etc]# mv resource.cfg-sample resource.cfg
root@nagios etc]# rm bigger.cfg-sample
```



Nagios: Server Configuration (4/5)

- Comment out all of the command definitions in your minimal.cfg file as these check commands are already defined in checkcommands.cfg to avoid the double reference

- Also change the below line in the above file to avoid the service reporting Total Processes UNKNOWN error on the web UI

```
command_line    $USER1$/check_procs -w $ARG1$ -c $ARG2$ -  
s $ARG3$
```

to

```
command_line    $USER1$/check_procs -w $ARG1$ -c $ARG2$
```

- Modify the /usr/local/nagios/etc/nagios.cfg file to set the "check_external_commands" to "1"



Nagios: Server Configuration (5/5)

- Restart Apache

```
root@nagios etc]# service httpd restart
```

- Test your Nagios configuration

```
root@nagios etc]# /usr/local/nagios/bin/nagios -v  
/usr/local/nagios/etc/nagios.cfg
```

- Start the Nagios service by

```
root@nagios etc]# service nagios start
```

- Navigate to <http://servername/nagios> and use nagiosadmin account
- You will only see Nagios monitoring the local host.



Nagios NRPE: Client Installation (1/2)

- Acquire the following latest packages from <http://www.nagios.org/download/>
 - nrpe-2.5.2.tar.gz
 - nagios-plugins-1.4.5.tar.gz
- Make a directory for placing the packages that you download:
root@nagiosclient ~]# **mkdir /root/nagiosinstall**
- Make a directory called “nagios” for the installation of the client:
root@nagiosclient ~]# **mkdir /usr/local/nagios**
- Unzip the nrpe-2.5.2.tar.gz file
root@nagiosclient ~]# **cd /root/nagiosinstall**
root@nagiosclient nagiosinstall]# **tar -xzvf nrpe-2.5.2.tar.gz**
- Configure and compile the nrpe client
root@nagiosclient nagiosinstall]# **cd nrpe-2.5.2**
root@nagiosclient nrpe-2.5.2]# **./configure -enable-command-args**
root@nagiosclient nrpe-2.5.2]# **make all**
- Copy the check_nrpe plugin from nrpe-2.5.2/src on NRPE client to /usr/local/nagios on your Nagios Server
root@nagiosclient nrpe-2.5.2]# **scp /root/nagiosinstall/nrpe-2.5.2/src/check_nrpe nagios:/usr/local/nagios/libexec**



Nagios NRPE: Client Installation (2/2)

- Copy the nrpe and nrpe.cfg files to /usr/local/nagios

```
root@nagiosclient nrpe-2.5.2]# cp ./src/nrpe /usr/local/nagios
root@nagiosclient nrpe-2.5.2]# cp ./sample-config/nrpe.cfg
/usr/local/nagios/
```

- Extract the nagios-plugins-1.4.3.tar.gz package

```
root@nagiosclient nrpe-2.5.2]# cd /root/nagiosinstall/
root@nagiosclient nagiosinstall]# tar-xzvf nagio-plugins-1.4.5
```

- Configure and compile the Nagios Plug-ins

```
root@nagiosclient nagiosinstall]# cd nagios-plugins-1.4.5
root@nagiosclient nagios-plugins-1.4.5]# ./configure
root@nagiosclient nagios-plugins-1.4.5]# make
root@nagiosclient nagios-plugins-1.4.5]# make install
```




Nagios NRPE: Client Configuration

- Open the `/usr/local/nagios/nrpe.cfg` file and change the line from `'dont_blame_nrpe=0'` to `'dont_blame_nrpe=1'`
- In the command section comment out all **unnecessary** tests.

```
command[check_local_disk]=/usr/local/nagios/libexec/check_disk -w $ARG1$ -c
  $ARG2$ -p $ARG3$
command[check_local_users]=/usr/local/nagios/libexec/check_users -w $ARG1$ -c
  $ARG2$
command[check_local_load]=/usr/local/nagios/libexec/check_load -w $ARG1$ -c
  $ARG2$
command[check_local_procs]=/usr/local/nagios/libexec/check_procs -w $ARG1$ -c
  $ARG2$
command[check_ping]=/usr/local/nagios/libexec/check_ping -H $ARG1$ -w $ARG2$
  -c $ARG3$ -p 5
```
- Make the user account and set the permission on the directory where you installed the NRPE client to

```
root@nagiosclient ~]# useradd nagios
root@nagiosclient ~]# chown -R nagios /usr/local/nagios
```
- Start the NRPE client

```
root@nagiosclient ~]# /usr/local/nagios/nrpe -c /usr/local/nagios/nrpe.cfg -d
```



Nagios NRPE: Server Configuration (1/2)

- Add the following to the checkcommand.cfg file on your Nagios Server

```
define command{
    command_name      check_nrpe
    command_line      /usr/local/nagios/libexec/check_nrpe -H
                    $HOSTADDRESS$ -c $ARG1$ -a $ARG2$ $ARG3$ $ARG4$
}
```

- Add to `hosts` section of /usr/local/nagios/minimal.cfg file

```
define host{
    use                generic-host                ; host template
    host_name          nagiosclient
    alias              nagiosclient
    address            1.2.3.4
    check_command      check-host-alive
    max_check_attempts 10
    check_period       24x7
    notification_interval 120
    notification_period 24x7
    notification_options d,r
    contact_groups     admins
}
```



Nagios NRPE: Server Configuration (1/2)

- Add the services to the 'services' section in /usr/local/nagios/minimal.cfg file, e.g.

```
define service{
    use                generic-service ; service template
    host_name          nagiosclient
    service_description PING
    is_volatile        0
    check_period       24x7
    max_check_attempts 4
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     admins
    notification_options w,u,c,r
    notification_interval 960
    notification_period 24x7
    check_command       check_nrpe!check_local_disk!20%!10%!/
}
```

- Restart the nagios service

```
root@nagios nagios-plugins-1.4.3]# service nagios restart
```

- Within a few minutes the Nagios Client should be Reported in the Nagios Server

- Troubleshooting:

```
root@nagios nagios-plugins-1.4.3]# /usr/local/nagios/bin/nagios -v  
/usr/local/nagios/etc/nagios.cfg
```

- It will tell you which file and what line nagios has a problem with when it won't run



Developing Nagios Plugins (1/2)

- Nagios plugins are standalone executables:
 - written in C, shell, perl, python, etc.
- Refer to the plug-in development guidelines
 - <http://nagiosplug.sourceforge.net/developer-guidelines.html>
- Nagios will only grab the **first line** of text from STDOUT
 - Stay within 80 characters
 - This will be used for text messages or paging
 - All ASGC plugins write result in **log file** for additional error messages
- Testing plugin
 - Add `-v` option for increased verbosity
 - Create unit test to simulate failures when the don't exist



Developing Nagios Plugins (2/2)

- Return Codes:
 - 0: OK
 - 1: Warning
 - 2: Critical
 - 3: Unknown – low level internal plugin errors (invalid arguments)
- Standard Options
 - List of standard options to give nagios plugins a more consistent interface
 - -H hostname, -t timeout, ...etc.
 - <http://nagiosplug.sourceforge.net/developer-guidelines.html#AEN304>
- Document Plugin
 - List user requirements for plugins
 - Tests executed by plugin
 - Specify plugin arguments and usage information



Nagios Plugins from ASGC (1/2)

- `init_vomsproxy`
 - Checks voms-proxy-init by creating a proxy on the Nagios host for GRID access
- `check_CE`
 - Checks globus-job-run by issuing job request to CE host to test functionality
- `check_GridFTP`
 - Checks functionality of GRID ftp services for given host by copying a test file and then deleting it
- `check_LFC`
 - Checks GRID Information Provider
 - Checks Catalog functionality
 - Checks copy-register (lcg-cr) functionality



Nagios Plugins from ASGC (2/2)

- check_SRM
 - Checks functionality of SRM services for specified host by copying a test file and then deleting it
- check_GStatUpdate
 - check if GStat is being updated on a timely basis
- check_HostCert
 - Check if the host public certificate is valid against the trusted CAs
 - Check if host certificate is about to expire



NRPE Plugins from ASGC

- **check_TimeSync**
 - Uses the ntpdate program to obtain the date and time for the given NTP server query
 - Generate an alert if time offset is above one of the warning and critical threshold values
 - If time is not in sync, then GSI security will fail
- **check_CApkg**
 - Checks to see if CA packages are up-to-date



Installing ASGC Plugins on Nagios Server

- Installation and Configuration on the Nagios server
 - Installation of UI software
 - Copy plugin into the /usr/local/nagios/libexec on Nagios server
 - Modify the necessary permissions and owners to run the plugin

```
root@nagios ~]# cd /usr/local/nagios/libexec
root@nagios libexec]# chmod 755 check_CE.py
root@nagios libexec]# chown nagios.nagios check_CE.p
```
 - Modify /usr/local/nagios/etc/checkcommands.cfg file to define the command

```
define command{
    command_name    check-CE
    command_line    python $USER1$/check_CE.py -g $ARG1$ -p $USER4$
                    -H $HOSTADDRESS$
}
```
 - Add the service to the 'services' section in /usr/local/nagios/minimal.cfg file

```
define service{
    use                checks
    host_name          ce-host-1
    service_description CE-chk
    check_command      check-CE!dteam
}
```



Installing ASGC Plugins on NRPE Client

- The following ASGC plugins (implemented in Python) are currently available
`check_TimeSync.py` `check_CApkg.py` `check_HostCert.py`
- Installation and Configuration on the NRPE client
 - Copy plugin into the `/usr/local/nagios/libexec` on NRPE client
 - Modify the necessary permissions and owners to run the plugin

```
root@nagiosclient ~]# cd /usr/local/nagios/libexec
root@nagiosclient libexec]# chmod 755 check_TimeSync.py
root@nagiosclient libexec]# chown nagios.nagios check_TimeSync.py
```
 - Modify `/usr/local/nagios/nrpe.cfg` file to define the command line

```
command[check_TimeSync]=python /usr/local/nagios/libexec/check_TimeSync.py -T $ARG1$
-w $ARG2$ -c $ARG3$
```
- Configuration on the Nagios server
 - Add the service to the 'services' section in `/usr/local/nagios/minimal.cfg` file

```
define service{
    use                checks
    host_name          nagiosclient
    service_description TimeSync-chk-nagiosclient
    check_command      check_nrpe!5666!check_TimeSync!140.109.98.230!30!120
}
```



Plugin Troubleshooting

- Service check timed out
 - Nagios plugin:
 - reset the `service_check_timeout` value on all service checks that run (`nagios.cfg`)
 - NRPE plugin:
 - reset the `check_nrpe -t` timeout to more seconds to see if it goes away (`checkcommands.cfg` or)
 - Wrong environment variables lead to the wrong path to use for SRM checks
- Grid ftp service checking failed on TW-FTT DPM hosts that reported the error message about processing certificate
 - Issue with voms proxy
 - allows you to create proxies with long lifetimes
 - but the extension information only shows 24 hours
 - Make the lifetimes of proxy to be less than 24 hours and then the problem goes away
- Proxy problems
 - Proxy is not valid long enough (3 hours) to run globus jobs for CE checking
 - Re-init proxy when life time is less than or equal to 3 hours
 - Unsymmetrical system time between checked host and Nagios host



SMS System

- Short Message Service (SMS) can send and receive short messages through GSM modems or mobile phones
- Using SMS service for Nagios contact notifications when service or host problems occur
 - Properly set the thresholds for notifications to send sms with nagios
- Sending SMS with Nagios is based on the misccommands.cfg, you have to define a command, which talks to your sms-notification-software such as sendsms or sms_client

- For using sendsms you can use the following:

```
define command{
  command_name notify-by-sms
  command_line /usr/local/bin/sendsms $CONTACTPAGER$ '$NOTIFICATIONTYPE$:
                $HOSTNAME$: $SERVICEDESC$ is $SERVICESTATE$ ($OUTPUT$)'
}
define command{
  command_name host-notify-by-sms
  command_line /usr/local/bin/sendsms $CONTACTPAGER$ '$NOTIFICATIONTYPE$:
                $HOSTNAME$ is $HOSTSTATE$ ($OUTPUT$)'
}
```

- 24x7 operations centers can utilize Nagios with SMS to manage grid resources on a more effective and efficient way



Thanks for Your Attention

Current Network Status
 Last Updated: Sat Dec 2 09:59:54 UTC 2006
 Updated every 90 seconds
 Nagios® - www.nagios.org
 Logged in as ops

Host Status Totals

Up	Down	Unreachable	Pending
51	0	0	2
All Problems		All Types	
0		53	

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
60	0	0	2	0
All Problems		All Types		
2		62		

Status Summary For All Service Groups

Service Group	Host Status Totals	Service Status Totals
Grid_Core_Service (Grid_Core_Service)	10 UP	10 OK 2 CRITICAL
IT_service (IT_service)	5 UP	5 OK
sc_castor (SC_CASTOR)	11 UP	14 OK
twgrid_T1_site (T1_resource)	2 UP	10 OK
TW_FTT_Site (Taiwan_FTT_Site)	12 UP	14 OK



Reference Links

- Download Nagios
 - <http://www.nagios.org/download/>
- Nagios Documentation
 - <http://www.nagios.org/docs/>
- Plug-in development guidelines
 - <http://nagiosplug.sourceforge.net/developer-guidelines.html>
- Nagios Screenshots
 - <http://www.nagios.org/about/screenshots.php>
- Nagios FAQ
 - <http://www.nagios.org/faqs/>
- The 3rd Party Plugin Repository
 - <http://www.nagiosexchange.org/>