10 Steps: MySQL Monitoring through Nagios: Install & Configure

Nagios is a powerful monitoring system and here we will learn how to monitor MySQL through Nagios. We will be installing Nagios, required plugins and configuring it to monitor MySQL Database Server.

Let's unleash the power step by step:

Installing and configuring Nagios

Step-1 : Install required stuff:

```
yum install httpd
yum install gcc
yum install glibc*
yum install gd*
```

Step-2 :Create Nagios user account and group

```
useradd nagios
passwd nagios
groupadd nagcmd
usermod -G nagcmd nagios
usermod -G nagcmd apache
```

Step-3: Downloads:

```
Create directory:
mkdir NagiosSetup
cd NagiosSetup
```

Download nagios

```
wget -X Get "http://sourceforge.net/projects/nagios/files/nagios-
3.x/nagios-3.2.1/nagios-3.2.1.tar.gz/download"
```

Download Nagios Plugins:

```
wget -X Get
"http://sourceforge.net/projects/nagiosplug/files/nagiosplug/1.4.15/nag
ios-plugins-1.4.15.tar.gz/download"
```

Step-4: Install Nagios

```
tar -xzvf nagios-3.2.1.tar.gz
cd nagios-3.2.1
./configure --with-command-group=nagcmd
make all
make install
make install-config
make install-commandmode
make install-init
chkconfig --add nagios
```

[If you miss step "make install-init" you may get:: error reading information on service nagios: No such file or directory]

Configure Nagios Web Interface:

```
make install-webconf
htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

[specify password for nagios admin]

Step-5: Install plugins

```
tar xvf nagios-plugins-1.4.11.tar.gz
cd nagios-plugins-1.4.11
./configure --with-nagios-user=nagios --with-nagios-group=nagios
make
make install
```

Step-6: Verify Installation, Starting nagios for the first time

```
service nagios start
```

Browse: http://localhost/nagios

Here if you get Error:
"You don't have permission to access /nagios/ on this server."

Check /etc/httpd/conf/httpd.conf for DirectoryIndex.
If it's not having index.php add it as follows:

```
vi /etc/httpd/conf/httpd.conf
DirectoryIndex index.php index.html index.html.var
```

Make sure you do restart apache(httpd) and nagios every time you change the config file.
You must have php installed.

Monitoring MySQL:

Step-7: Download, Extract and install the MySQL Plugin:

```
wget http://labs.consol.de/wp-
content/uploads/2010/10/check_mysql_health-2.1.3.tar.gz

tar -zxvf check_mysql_health-2.1.3.tar.gz
cd check_mysql_health-2.1.3
./configure --prefix=/usr/local/nagios --with-nagios-user=nagios --
with-nagios-group=nagios --with-perl=/usr/bin/perl
make
make install
```

Step-8: Create database user:

```
grant all privileges on *.* to 'nagios'@'localhost' identified by
'nagios';
```

Step-9: Provide email address for nagiosadmin:

[Change contacts.cfg file accordingly.]

```
vi /usr/local/nagios/etc/objects/contacts.cfg

define contact{
contact_name nagiosadmin ; Short name of user
use generic-contact ; Inherit default values from generic-contact
template (defined above)
alias Kedar ; Full name of user
email kedar@nitty-witty.com ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS
******
}
```

Step-10: Configuring Nagios to Monitor MySQL Server

```
vi /usr/local/nagios/etc/nagios.cfg
```

add following line:

```
cfg_file=/usr/local/nagios/etc/objects/mysqlmonitoring.cfg
```

Define check_mysql_health command as follows:

```
vi /usr/local/nagios/etc/objects/commands.cfg

define command{
command_name check_mysql_health
command_line $USER1$/check_mysql_health -H $ARG4$ --username $ARG1$ --
password $ARG2$ --port $ARG5$ --mode $ARG3$
}
```

Enter services to be monitored in mysqlmonitoring.cfg:

```
vi /usr/local/nagios/etc/objects/mysqlmonitoring.cfg
```

Add:

```
define service{
use local-service
host_name localhost
service_description MySQL connection-time
check_command check_mysql_health!nagios!nagios!connection-
time!127.0.0.1!3306!
}

define service{
use local-service
host_name localhost
service_description MySQL slave-io-running
check_command check_mysql_health!nagios!nagios!slave-io-
running!127.0.0.1!3306!
}
```

```
define service{
use local-service
host_name localhost
service_description MySQL slave-sql-running
check_command check_mysql_health!nagios!nagios!slave-sql-
running!127.0.0.1!3306!
}
```

Here we've monitored 3 services: Connection-time, io thread and sql thread (replication) status. You can monitor more parameters described here:
http://labs.consol.de/nagios/check_mysql_health/

Note: Every time you change configuration file, verify before starting nagios using command:
`/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg`

Finally start nagios service and you're done with nagios installation and configuration for monitoring MySQL.
I hope you've found this useful.