**NAGIOS**

* It is a tool for actively monitoring services and devices in your server. It can support monitoring up to thousands of services and devices. It is the most used open source monitoring tool.
* It monitors servers and services associated with the servers. If these not working properly it sends an notifications in the form of mail for the person or a group.
* Normally, you need to write code to monitor the server. But, in nagios it uses plug-ins to check the specific servers and it has its own database to store the data. You can use the stored data to see the server status and errors. you can generate a report from this database.

**INSTALL NAGIOS**

* Download epel rpm package from internet, install it and do system update.
* Install Nagios = **yum** **install nagios nagios-plugins-all nagios-plugins-nrpe nrpe**.
* Install http and php for dashboard = **yum install httpd php**\*.
* Add nagios user to apache and nagios groups.

**Usermod –G nagios nagios.**

**Usermod –G nagios apache.**

* Now, create an nagios account to login to nagios web interface. By default, nagiosadmin is the default user to view nagios dashboard.

**htpasswd –c /etc/nagios/passwd nagiosadmin**

* When you enter the above command, it will ask you to set password for nagiosadmin.
* Start the apache server and nagios server.
* Go to browser, type **yourip/nagios**. It will ask for user name and password that we created earlier. Login with those credentials to see nagios dashboard.
* If it shows **”X not running”** error in nagios dashboard. Disable **selinux**.
* Go to /etc/nagios/objects dir and define all your hosta and services in localhost.cfg file.

**define host {**

**use linux-server**

**host\_name ip-172-31-23-85.ap-south-1.compute.internal**

**alias remote**

**address your ip.**

**}**

**define service {**

**use local-service**

**host\_name localhost**

**service\_description PING**

**check\_command check\_ping!100.0,20%!500.0,60%**

**check\_interval 2**

**}**

**define service {**

**use local-service**

**host\_name localhost**

**service\_description Root Partition**

**check\_command check\_local\_disk!20%!10%!/**

**}**

* You can also configure your hosts and services in separate cfg file for each host. After that you have to mention that file path in nagios.cfg file.
* To check errors in nagios configuration files = **/usr/sbin/nagios -v /etc/nagios/nagios.cfg**
* If you want use different username and password for web interface.
* Go to /**etc/nagios/objects/cgi.cfg** file and add your user name in the fileds where you see the username “nagiosadmin” .
* In /etc/nagios directory, there is a directory called **objects**. In this objects dir, all the supporting config files will be there.

In Nagios.cfg file,

**Status\_update interval** = time(sec) – to update status.dat file.

**Nagios\_user** – the effective user where nagios should run as.

**Nagios\_group** – the effective group that nagios should run as.

**Set log\_notifications** = 1 – it will log the notifications in nagios.log file.

* When we first run nagios, it checks the nagios.cfg file, using this file, nagios will read all the other config files written in this file and monitor based on our configuration.
* **Commands.cfg** – it contains the commands for all the checks. Means that, when we define a check to host, it will monitor host based on these commands as mentioned in commands.cfg file.
* **Templates.cfg** – it contains group of values used to define the hosts (or) services to monitor. You can create your own template and give the values based on your need. By these templates, we don’t need to give check values for every host; we can simply use a template.
* **Localhost.cfg** – we need to define the servers to monitor based on our config. We need to give the details of our hosts and template to use to monitor in this file. We need to define hostname, template, alias, host ip.

After defining the host, we have to define the services to the host. You can define those in the same file below few lines. There you have to mention hostname, checks, service description.

**MONITOR REMOTE SERVER**

* You can monitor remote server by using nagios **nrpe** plugin. You have to install nrpe in remote server and add the remote server configurations in nagios server.
* Nrpe plugin will fetch the remote server data to nagios server.
* Install nrpe plugin = **yum install nagios-plugins-all nagios-nrpe nrpe**.
* After installing, edit the **/etc/nagios/nrpe.cfg** file,

**port = 5666.**

**allow from = 127.0.0.1, server ip.**

**don't blame nrpe = 1**.

* If we configured everything correctly, try Start the nrpe service.
* To monitor remote host and services from nagios server,

**You need to download nrpe plugin in Nagios server.**

**Define commads in nrpe.cfg in remote server.**

* **yum install nagios-plugins-nrpe** = Install nrpe plugin in Nagios server.
* In remote server, open **/etc/nagios/nrpe.cfg** and Go to **MISC SYSTEM METRICS**.
* Uncomment the **service** commands, specify the **warning** and **critical** values.
* Restart the **nrpe** service.
* Go to Nagios server, check the remote services with nrpe plugin

**/usr/lib64/nagios/plugins/check\_nrpe –H remoteserverip –c check\_disk**

* Here, I am checking remote server’s disk usage. To monitor other services, specify those plugins in the place of disk plugin.
* If it shows you the output, it means nrpe plugin is working properly.
* Now, we have to define the nrpe command in command.cfg.

**define command{**

**command\_name check\_nrpe**

**command\_line $USER1$/check\_nrpe -H $HOSTADDRESS$ -c $ARG1$**

**}**

* Now, define your remote server host and services in your remote configuration file.
* In **/etc/nagios/objects dir**, Create a new cfg file, define remote hosts and services.
* For the check\_command add your nrpe command and your service plugins. So that nagios will know whether its remote server’s (or) local server’s configuration.

**check\_command check\_nrpe!check\_disk**

* You can add as many services as you want with the nrpe plugin to monitor remote hosts and services.
* Add a line in **nagios.cfg** for the remote host cfg file.
* Check for the errors and restart nagios server to update the web-interface with remote server.

**MONITOR CPU, RAM:**

* By default, nagios won’t provide any plugins for cpu and ram. As you can see in the nagios dashboard there would be no cpu and ram services.
* We have to download those **cpu** and **ram** **plugins** from the internet and define them to monitor.

[**https://raw.githubusercontent.com/justintime/nagios-plugins/master/check\_mem/check\_mem.p**l](https://raw.githubusercontent.com/justintime/nagios-plugins/master/check_mem/check_mem.pl) = download **ram** plugin from this url.

* You will get a perl script named **check\_mem.pl**. change the name to **check\_mem**. Give execute permission to this script.
* Move this plugin to **/usr/lib64/nagios/plugins** dir where othere plugins are placed.

**./check\_mem –f –w 30 –c 10** = to check whether plugin is working or not.

* Once it is working properly, you can monitor RAM of nagios server.
* Define a command for check\_mem plugin in commands.cfg.

**define command{**

**command\_name check\_mem**

**command\_line $USER1$/check\_mem -f -w $ARG1$ -c $ARG2$**

**}**

* Define service for check\_mem in localhost.cfg to monitor server RAM with nagios.

**define service{**

**use local-service**

**host\_name localhost**

**service\_description local RAM**

**check\_command check\_mem!20%!10%!/**

**}**

* Restart the nagios server to apply the changes. Now, in the nagios dashboard you can see the RAM service.
* To monitor remote server RAM with nagios server.
* Download the **check\_ram** plugin on remote server.
* Move this plugin to **/usr/sbin/nagios/plugins** dir where othere plugins are placed.
* Uncomment the **check\_mem** line in nrpe.cfg under **MISC SYSTEM METRICS**.

**command[check\_mem]=/usr/lib64/nagios/plugins/check\_mem**

* Add warning and critical values in memory line. Save the file and restart nrpe service.
* Go to nagios server, define service to monitor RAM of remote server. By default, we have to check remote hosts and services with nrpe plugin. We need to execute mem plugin along with nrpe plugin.

**define service{**

**use local-service**

**host\_name remote server hostname**

**service\_description REMOTE RAM**

**check\_command check\_nrpe!check\_mem**

**}**

* Once, defining the service restart the nagios server to apply the changes in dashboard.
* To monitor CPU with Nagios. Download plugin from this url.

[**http://lancet.mit.edu/mwall/projects/nagios/check\_cpu**](http://lancet.mit.edu/mwall/projects/nagios/check_cpu)

* Give execute permissions to the plugin and move this plugin where other plugins are placed.
* Execute this to check whether plugin is working or not = **./check\_cpu –w 10 –c 30**
* Warning value should be less than critical value for cpu.
* Once it is working properly, define a command for cpu in commands.cfg to monitor nagios server CPU.

**define command{**

**command\_name check\_mem**

**command\_line $USER1$/check\_mem -f -w $ARG1$ -c $ARG2$**

**}**

* Define service to monitor nagios server CPU in localhost.cfg

**define service{**

**use local-service**

**host\_name localhost**

**service\_description check\_cpu**

**check\_command check\_mem!10!20!/**

**}**

* To monitor remote server CPU with nagios server.
* Download the **check\_cpu** plugin on remote server.
* Move this plugin to **/usr/sbin/nagios/plugins** dir where othere plugins are placed.
* Uncomment the **check\_cpu** line in nrpe.cfg under **MISC SYSTEM METRICS**.

**command[check\_mem]=/usr/lib64/nagios/plugins/check\_mem**

* Add warning and critical values in cpu line. Save the file and restart nrpe service.
* Go to nagios server, define service to monitor CPU of remote server.

**define service{**

**use local-service**

**host\_name remote server hostname**

**service\_description REMOTE CPU**

**check\_command check\_nrpe!check\_CPU**

**}**

* Once, defining the service restart the nagios server to apply the changes in dashboard.

**MONITOR MYSQL SERVER**

* You can monitor mysql databse with nagios tool. You have to download mysql plugin and configure the plugin in nagios and create an user in mysql and give permissions to that user for accessing mysql from nagios.
* Login to mysql server and create a user named nagios with full access. And create user nagios again with anywhere access(%). Type % in the place of localhost.
* Go to **nagios exchange**, download mysql plugin = **check\_mysql\_health** .
* Go to extracted plugin dir, compile the code = **./configure, make** and **make install**.
* You have to install perl modules = **yum install perl-DBD-MySQL.**
* Go to commands.cfg file, paste this script at the end of file to monitor mysql.

**define command{**

**command\_name check\_mysql\_health**

**command-line $USER1$/check\_mysql\_health -H $ARG1$ --port $ARG2$ --username $ARG3$ --password $ARG4$ --mode $ARG5$**

**}**

* Go to **localhost.cfg**, define services for mysql, save the file check for errors and restart.
* **./check\_mysql\_health –H serverip –user mysql-user –password password –mode** **uptime** = To check mysql uptime.

**Define service {**

**Use local-service (template)**

**Host name localhost (or) server-ip**

**Service description Mysql Uptime**

**Check\_command check\_mysql\_health!server-ip!3306!db-user!db-pass!uptime**

**}**

* Now, mysql is available in nagios dashboard.
* If you want to check another service in mysql, copy the defined service and add the service name in the place of uptime.

**NOTIFICATIONS**

* Nagios will send notifications in any specific action occurs based on our configuration.
* There will be default templates for contacts in templates.cfg file to send mails at regular times, specific action occurs etc. You can use those templates to send notifications.
* By default, nagios check the hosts and services for every five minutes **check\_interval** **5**….(5=min). You can change in **templates.cfg**.
* In templates.cfg, edit your hosts and services templates and change check interval to your desired time and save it.
* Before defining contacts in nagios, make sure you have Installed and configured **postfix** mail server, where the nagios server running to send mails.
* **yum install postfix** = To install postfix.
* Install **telnet** to check the postfix.
* **telnet localhost 25**……to check whether postfix is working (or) not.
* If it shows output like this **220 ip-192-168-10-31.ap-south-1.compute.internal ESMTP Postfix**. Then, you have installed postfix successfully.
* Now you have to define contacts and contact groups in nagios.
* Go to /etc/nagios/objects/contacts.cfg.
* Define a contact under contact section.

**define contact{**

**contact\_name admins (unique)**

**use generic-contact (template to use)**

**alias alias for name**

**email emails to send notifications**

**}**

* Create as many contacts as you want.
* Define contactgroup under contactgroups section to add multiple contacts as a single group. So, if we assign notifications this group, all the users in this specific group will get notified when an specific action occurs.

**define contactgroup{**

**contactgroup\_name name of the group**

**alias alias name for the group**

**members contacts to add to this group**

**}**

* Go to localhost.cfg file (or) the file you have defined the hosts and services.
* Add **contact\_groups group-name** line in hosts and services sections you want to get notified.