DEVOPS – CI/CD NAGIOS

23-11-2020 to 27-11-2020

Venkatesh Reddy Madduri && ShankarPrasad

Agenda

- What is Continuous Monitoring
- What is Nagios
- Architecture
- Configuration
- Features
- Applications
- Hosts and Services
- Commands
- Checks and States
- Ports and Protocols
- Add-Ons / Plugins
- NRPE
- Additional Products
- Installation and Demo

What is Continuous Monitoring ?

Continuous monitoring starts when the deployment is done on the production servers. From then on, this stage is responsible to monitor everything happening. This stage is very crucial for the business productivity.

There are several benefits of using Continuous monitoring −

- It detects all the server and network problems.
- It finds the root cause of the failure.
- It helps in reducing the maintenance cost.
- It helps in troubleshooting the performance issues.
- It helps in updating infrastructure before it gets outdated.
- It can fix problems automatically when detected.
- It makes sure the servers, services, applications, network is always up and running.
- It monitors complete infrastructure every second.

What is Nagios and Why?

Nagios is an open-source continuous monitoring tool which monitors network, applications and servers. It can find and repair problems detected in the infrastructure, and stop future issues before they affect the end users. It gives the complete status of your IT infrastructure and its performance.

Why?

- It can monitor Database servers such as SQL Server, Oracle, Mysql, Postgres
- It gives application level information (Apache, Postfix, LDAP, Citrix etc.).
- Provides active development.
- Has excellent support form huge active community.
- Nagios runs on any operating system.
- It can ping to see if host is reachable.

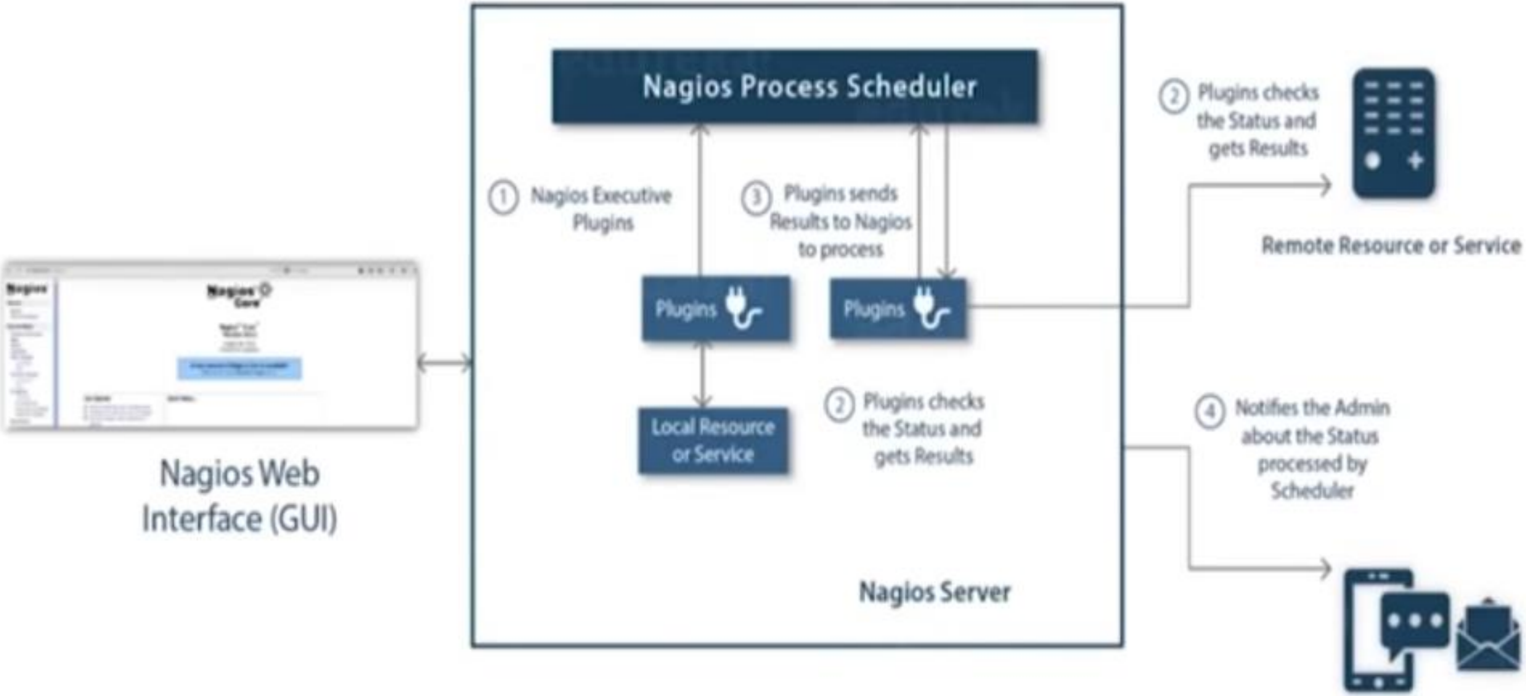
Architecture

The following points are worth notable about Nagios architecture −

- Nagios has server-agent architecture.
- Nagios server is installed on the host and plugins are installed on the remote hosts/servers which are to be monitored.
- Nagios sends a signal through a process scheduler to run the plugins on the local/remote hosts/servers.
- Plugins collect the data (CPU usage, memory usage etc.) and sends it back to the scheduler.
- Then the process schedules send the notifications to the admin/s and updates Nagios GUI.

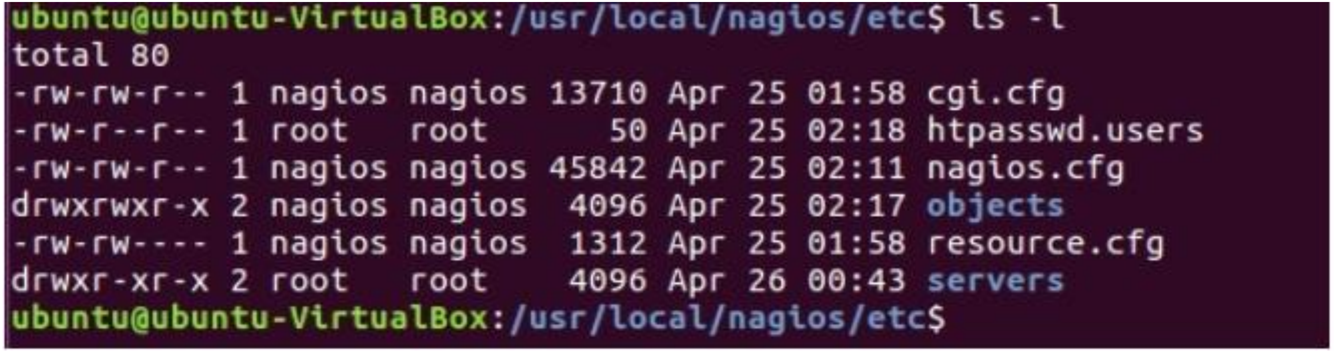
Architecture

Nagios®


Nagios Process Scheduler
① Nagios Executive Plugins
③ Plugins sends Results to Nagios to process
Plugins
Plugins
Local Resource or Service
② Plugins checks the Status and gets Results
Nagios Server
② Plugins checks the Status and gets Results
Remote Resource or Service
④ Notifies the Admin about the Status processed by Scheduler
Nagios Web Interface (GUI)
Nagios Core

Configuration Files

The configuration files of Nagios are located **in /usr/local/nagios/**etc. These files are shown in the screenshot given below –



```
ubuntu@ubuntu-VirtualBox:/usr/local/nagios/etc$ ls -l
total 80
-rw-rw-r-- 1 nagios nagios 13710 Apr 25 01:58 cgi.cfg
-rw-r--r-- 1 root   root      50 Apr 25 02:18 htpasswd.users
-rw-rw-r-- 1 nagios nagios 45842 Apr 25 02:11 nagios.cfg
drwxrwxr-x 2 nagios nagios  4096 Apr 25 02:17 objects
-rw-rw---- 1 nagios nagios  1312 Apr 25 01:58 resource.cfg
drwxr-xr-x 2 root   root    4096 Apr 26 00:43 servers
ubuntu@ubuntu-VirtualBox:/usr/local/nagios/etc$
```

Let us understand the importance of each file now

Configuration Files

nagios.cfg

- This is the main configuration file of Nagios core.
- File contains the location of log file of Nagios, hosts and services state update interval, lock file and status.dat file.
- Nagios users and groups on which the instances are running are defined in this file.
- It has path of all the individual object config files like commands, contacts, templates etc.

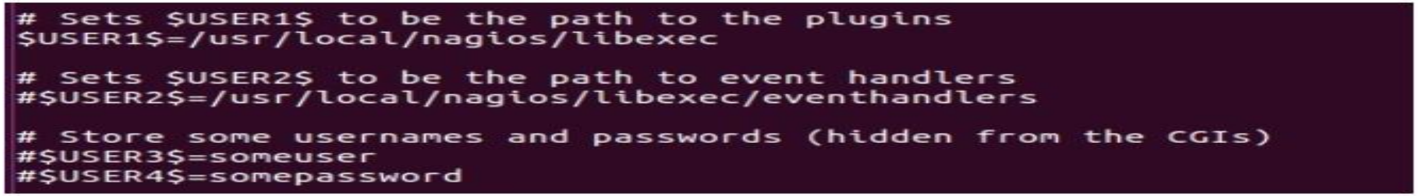
cgi.cfg [Common Gateway Interface]

- It tells the CGIs where to find the main configuration file.
- The CGIs will read the main and host config files for any other data they might need.
- It contains all the user and group information and their rights and permissions.
- It also has the path for all frontend files of Nagios.

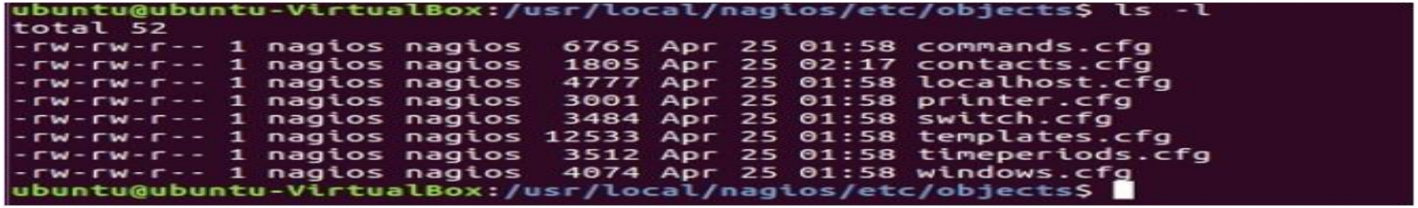
Configuration Files

resource.cfg

- We can define $USERx$ macros in this file, which can in turn be used in command definitions in your host config file(s). $USERx$ macros are useful for storing sensitive information such as usernames, passwords, etc.
- They are also handy for specifying the path to plugins and event handlers - if you decide to move the plugins or event handlers to a different directory in the future, you can just update one or two $USERx$ macros, instead of modifying a lot of command definitions. Resource files may also be used to store configuration directives for external data sources like MySQL



```
# Sets $USER1$ to be the path to the plugins
$USER1$=/usr/local/nagios/libexec

# Sets $USER2$ to be the path to event handlers
#$USER2$=/usr/local/nagios/libexec/eventhandlers

# Store some usernames and passwords (hidden from the CGIs)
#$USER3$=someuser
#$USER4$=somepassword
```



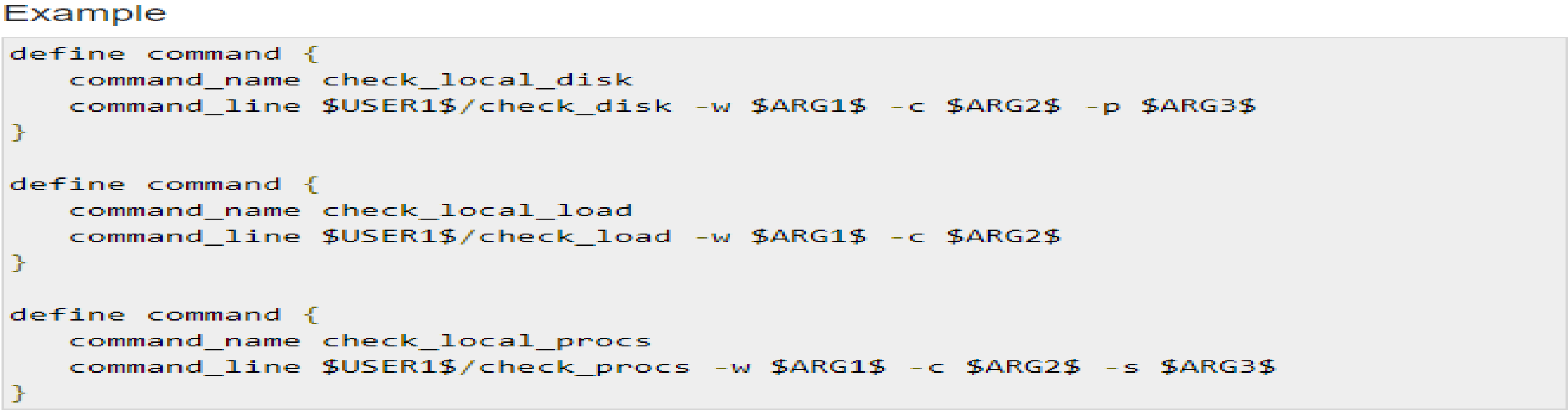
```
ubuntu@ubuntu-VirtualBox:/usr/local/nagios/etc/objects$ ls -l
total 52
-rw-rw-r-- 1 nagios nagios  6765 Apr 25 01:58 commands.cfg
-rw-rw-r-- 1 nagios nagios  1805 Apr 25 02:17 contacts.cfg
-rw-rw-r-- 1 nagios nagios  4777 Apr 25 01:58 localhost.cfg
-rw-rw-r-- 1 nagios nagios  3001 Apr 25 01:58 printer.cfg
-rw-rw-r-- 1 nagios nagios  3484 Apr 25 01:58 switch.cfg
-rw-rw-r-- 1 nagios nagios 12533 Apr 25 01:58 templates.cfg
-rw-rw-r-- 1 nagios nagios  3512 Apr 25 01:58 timeperiods.cfg
-rw-rw-r-- 1 nagios nagios  4074 Apr 25 01:58 windows.cfg
ubuntu@ubuntu-VirtualBox:/usr/local/nagios/etc/objects$ 
```

- The configuration files inside objects directory are used to define **commands, contacts, hosts, services** etc.

Configuration Files

commands.cfg

- This config file provides you with some example command definitions that we can refer in host, service, and contact definitions.
- These commands are used to check and monitor hosts and services.
- We can run these commands locally on a Linux console where we will also get the output of the command you run.



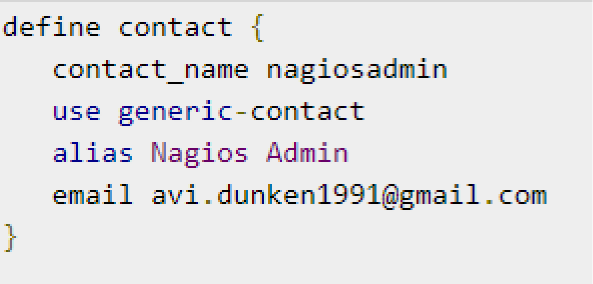
Example

```
define command {
    command_name check_local_disk
    command_line $USER1$/check_disk -w $ARG1$ -c $ARG2$ -p $ARG3$
}

define command {
    command_name check_local_load
    command_line $USER1$/check_load -w $ARG1$ -c $ARG2$
}

define command {
    command_name check_local_procs
    command_line $USER1$/check_procs -w $ARG1$ -c $ARG2$ -s $ARG3$
}
```

Configuration Files

contacts.cfg

This file contains contacts and groups information of Nagios. By default, one contact is already present Nagios admin.



```
define contact {
   contact_name nagiosadmin
   use generic-contact
   alias Nagios Admin
   email avi.dunken1991@gmail.com
}
```

Features

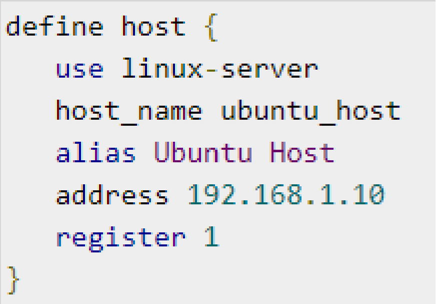
- Nagios Core is open source, hence free to use.
- Powerful monitoring engine which can scale and manage 1000s of hosts and servers.
- Comprehensive web dashboard giving the visibility of complete network components and monitoring data.
- It has multi-tenant capabilities where multiple users have access to Nagios dashboard.
- It has extendable architecture which can easily integrate with third-party applications with multiple APIs.
- Nagios has a very active and big community with over 1 million + users across the globe.
- Fast alerting system, sends alerts to admins immediately after any issue is identified.
- Multiple plugins available to support Nagios, custom coded plugins can also be used with Nagios.
- It has good log and database system storing everything happening on the network with ease.
- Proactive Planning feature helps to know when it’s time to upgrade the infrastructure.

Applications

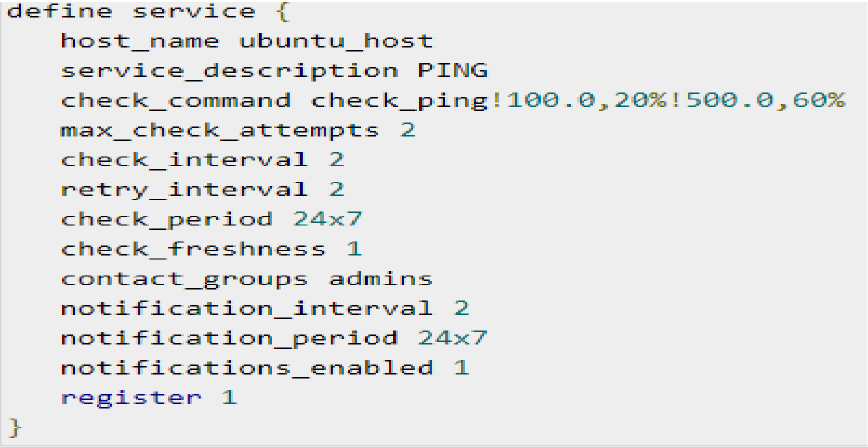
- Monitor host resources such as disk space, system logs etc.
- Monitor network resources – http, ftp, smtp, ssh etc.
- Monitor log files continuously to identify infra-issue.
- Monitor windows/linux/unix/web applications and its state.
- Nagios Remote Plugin Executer (NRPE) can monitor services remotely.
- Run service checks in parallel.
- SSH or SSL tunnels can also be used for remote monitoring.
- Send alerts/notifications
- via email, sms, pager of any issue on infrastructure
- Recommending when to upgrade the IT infrastructure.

Host and Services

- Nagios is the most popular tool which is used to monitor hosts and services running in your IT infrastructure.
- Hosts and service configurations are the building blocks of Nagios Core.
- Host is just like a computer; it can be a physical device or virtual.
- Services are those which are used by Nagios to check something about a host.
- We can create a host file inside the server directory of Nagios and mention the host and service definitions.



```
define host {
   use linux-server
   host_name ubuntu_host
   alias Ubuntu Host
   address 192.168.1.10
   register 1
}
```



```
define service {
    host_name ubuntu_host
    service_description PING
    check_command check_ping!100.0,20%!500.0,60%
    max_check_attempts 2
    check_interval 2
    retry_interval 2
    check_period 24x7
    check_freshness 1
    contact_groups admins
    notification_interval 2
    notification_period 24x7
    notifications_enabled 1
    register 1
}
```

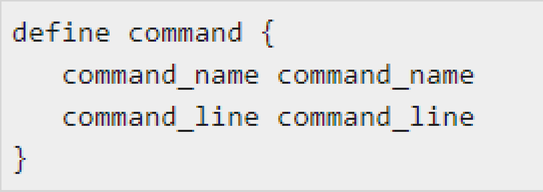
Commands

- A command definition defines a command.

- Commands include service checks, service notifications, service event handlers, host checks, host notifications, and host event handlers.

- Command definitions for Nagios are defined in commands.cfg file.

The following is the format for defining of a Command −



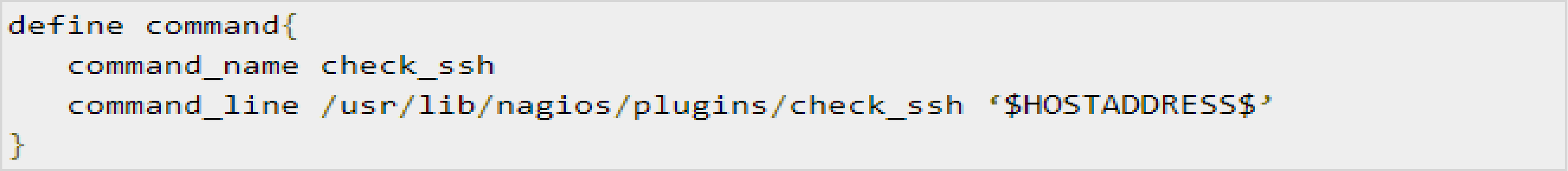
```
define command {
   command_name command_name
   command_line command_line
}
```

Command name − This directive is used to identify the command. The definitions of contact, host, and service is referenced by command name.

Command line − This directive is used to define what is executed by Nagios when the command is used for service or host checks, notifications, or event handlers.

Commands

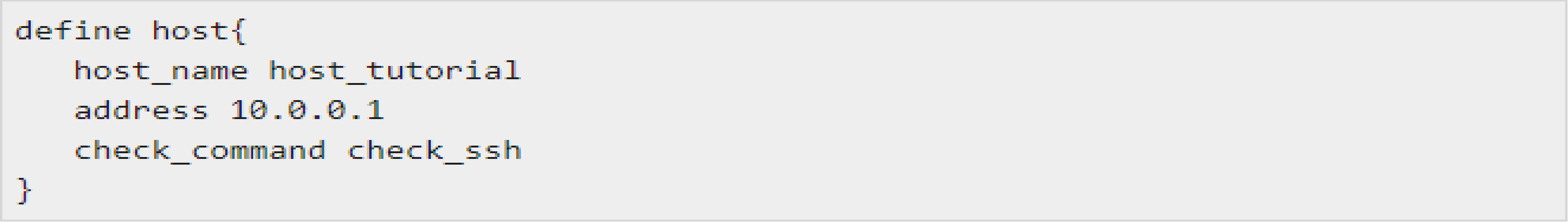
Example



```
define command{
   command_name check_ssh
   command_line /usr/lib/nagios/plugins/check_ssh ‘$HOSTADDRESS$’
}
```

This command will execute the plugin – /usr/libl/nagios/plugins/check_ssh with 1 parameter : '$HOSTADDRESS$'

A very short host definition that would use this check command could be similar to the one shown here –

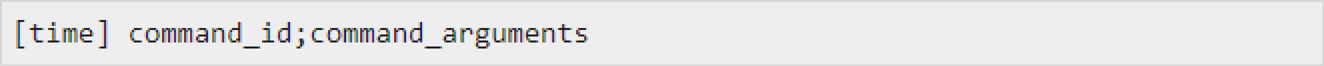


```
define host{
   host_name host_tutorial
   address 10.0.0.1
   check_command check_ssh
}
```

Commands

- ➢ We can run external commands in Nagios by adding them to commands file which is processed by Nagios daemon periodically.
- ➢ With External commands we can achieve lot many checks while Nagios is running. We can temporarily disable few checks, or force some checks to run immediately, disable notifications temporarily etc.

The following is the syntax of external commands in Nagios that must be written in command file −



```
[time] command_id;command_arguments
```

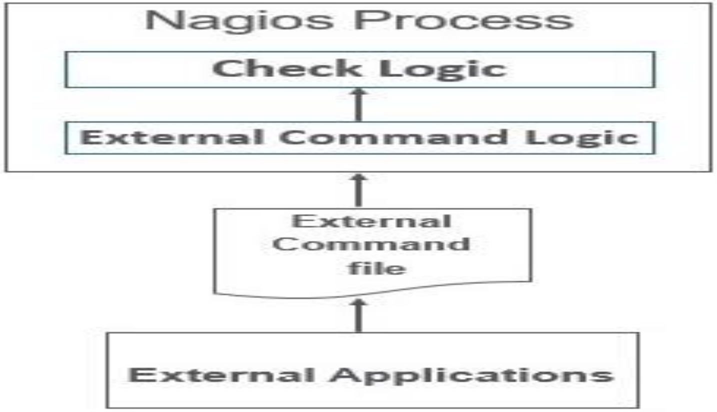
- ➢ We can also check out the list of all external commands that can be used in Nagios here https://assets.nagios.com/downloads/nagioscore/docs/externalcmds/

Checks and Status

Passive **Checks**

Passive checks are performed by external processes and the results are given back to Nagios for processing.

An external application checks the status on hosts/services and writes the result to External Command File. When Nagios daemon reads external command file, it reads and sends all the passive checks in the queue to process them later. Periodically when these checks are processed, notifications or alerts are sent depending on the information in check result.



Checks and Status

Nagios stores the status of the hosts and services it is monitoring to determine if they are working properly or not. There would be many cases when the failures will happen randomly and they are temporary; hence Nagios uses states to check the current status of a host or service.

There are two types of states −

- ➢ Soft state
- ➢ Hard state

Soft state

When a host or service is down for a very short duration of time and its status is not known or different from previous one, then soft states are used. The host or the services will be tested again and again till the time the status is permanent.

Hard State

When **max_check_attempts** is executed and status of the host or service is still not OK, then hard state is used. Nagios executes event handlers to handle hard states.

Protocols and Ports

Protocols

The default protocols used by Nagios are as given under −

- http(s), ports 80 and 443 − The product interfaces are web-based in Nagios. Nagios agents can use http to move data.
- snmp, ports 161 and 162 − snmp is an important part of network monitoring. Port 161 is used to send requests to nodes and post 162 is used to receive results.
- ssh, port 22 − Nagios is built to run natively on CentOS or RHEL Linux. Administrator can login into Nagios through SSH whenever they feel to do so and perform checks.

Ports

The Default ports used by common Nagios Plugins are as given under −

- Butcheck_nt (nsclient++) 12489 , NRPE 5666, NSCA 5667, NCPA 5693, MSSQL 1433,MySQL 3306

  PostgreSQL 5432,MongoDB 27017, 27018,OracleDB 1521,Email (SMTP) 25, 465, 587,

  WMI 135, 445 / additionaldynamically-assigned ports in 1024-1034 range

Protocols and Ports

Protocols

The default protocols used by Nagios are as given under −

- http(s), ports 80 and 443 − The product interfaces are web-based in Nagios. Nagios agents can use http to move data.
- snmp, ports 161 and 162 − snmp is an important part of network monitoring. Port 161 is used to send requests to nodes and post 162 is used to receive results.
- ssh, port 22 − Nagios is built to run natively on CentOS or RHEL Linux. Administrator can login into Nagios through SSH whenever they feel to do so and perform checks.

Ports

The Default ports used by common Nagios Plugins are as given under −

- Butcheck_nt (nsclient++) 12489 , NRPE 5666, NSCA 5667, NCPA 5693, MSSQL 1433,MySQL 3306

  PostgreSQL 5432,MongoDB 27017, 27018,OracleDB 1521,Email (SMTP) 25, 465, 587,

  WMI 135, 445 / additionaldynamically-assigned ports in 1024-1034 range

Add-ons and Plug-ins

- ➢ Plugins helps to monitor databases, operating systems, applications, network equipment, protocols with Nagios.
- ➢ Plugins are compiled executables or script (Perl or non-Perl) that extends Nagios functionality to monitor servers and hosts.
- ➢ Nagios will execute a Plugin to check the status of a service or host.
- ➢ Nagios can be compiled with support for an embedded Perl interpreter to execute Perl plugins.

Types of Nagios Plugins

Nagios has the following plugins available in it −

- **Official Nagios Plugins** − There are 50 official Nagios Plugins. Official Nagios plugins are developed and maintained by the official Nagios Plugins Team.
- **Community Plugins** − There are over 3,000 third party Nagios plugins that have been developed by hundreds of Nagios community members.
- **Custom Plugins** − We can also write our own Custom Plugins. There are certain guidelines that must be followed to write Custom Plugins.

Add-ons and Plug-ins

Guidelines for Writing Custom Nagios Plugins

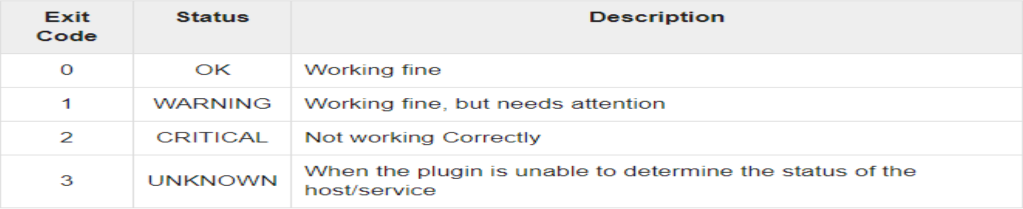
- While writing custom plugin in Nagios, you need to follow the guidelines given below −
- Plugins should provide a "-V" command-line option (verify the configuration changes)
- Print only one line of text
- Print the diagnostic and only part of the help message
- Network plugins use DEFAULT_SOCKET_TIMEOUT to timeout
- "-v", or "--verbose“ is related to verbosity level
- "-t" or "--timeout" (plugin timeout);
- "-w" or "--warning" (warning threshold);
- "-c" or "--critical" (critical threshold);
- "-H" or "--hostname" (name of the host to check)

Add-ons and Plug-ins

Guidelines for Writing Custom Nagios Plugins

- While writing custom plugin in Nagios, you need to follow the guidelines given below −
- Plugins should provide a "-V" command-line option (verify the configuration changes)
- Print only one line of text
- Print the diagnostic and only part of the help message
- Network plugins use DEFAULT_SOCKET_TIMEOUT to timeout
- "-v", or "--verbose“ is related to verbosity level
- "-t" or "--timeout" (plugin timeout);
- "-w" or "--warning" (warning threshold);
- "-c" or "--critical" (critical threshold);
- "-H" or "--hostname" (name of the host to check)

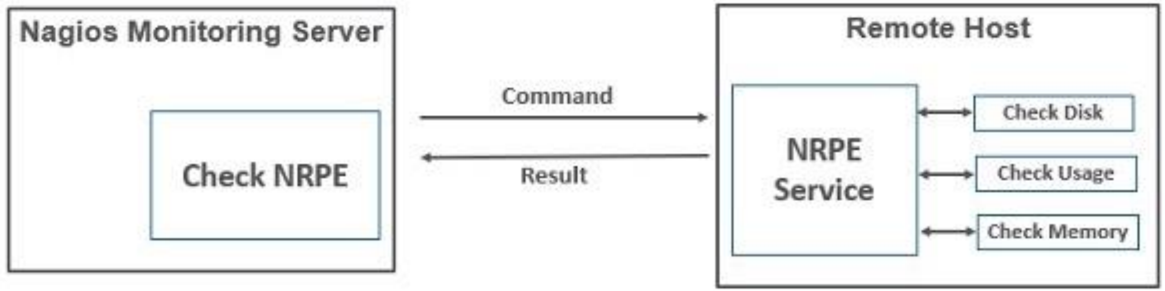
Multiple Nagios plugin run and perform checks at the same time, for all of them to run smoothly together, Nagios plugin follow a status code. The table given below tells the exit code status and its description −



Exit Code	Status	Description
0	OK	Working fine
1	WARNING	Working fine, but needs attention
2	CRITICAL	Not working Correctly
3	UNKNOWN	When the plugin is unable to determine the status of the host/service

NRPE – Nagios Remote Plugin Executor

The Nagios daemon which run checks on remote machines in NRPE (Nagios Remote Plugin Executor). It allows you to run Nagios plugins on other machines remotely. You can monitor remote machine metrics such as disk usage, CPU load etc. It can also check metrics of remote windows machines through some windows agent addons.



Products

Nagios Core

It is the core on monitoring IT infrastructure. Nagios XI product is also fundamentally based on Nagios core. Whenever there is any issue of failure in the infrastructure, it sends an alert/notification to the admin who can take the action quickly to resolve the issue. This tool is absolutely free.

Nagios XI

It provides monitoring for complete IT infrastructure components like applications, services, network, operating systems etc. It gives a complete view of your infrastructure and business processes. The GUI is easily customizable giving the used flexibility.

The standard edition of this tool costs $1995 and enterprise edition costs $3495.

Products

Nagios Log Server

It makes searching of log data very simple and easy. It keeps all the log data at one location with high availability setup. It can easily send alerts if any issue is found in the log data. It can scale to 1000s of severs giving more power, speed, storage, and reliability to your log analysis platform.

The price of this tool depends on the number of instances - 1 Instance $3995, 2 Instances $4995, 3 Instances $5995, 4 Instances $6995, 10 Instances $14995.

Nagios Fusion

This product provides a centralized view of complete monitoring system. With Nagios Fusion, we can scan setup separate monitoring servers for different geographies. It can be easily integrated with Nagios XI and Nagios core to give the complete visibility of the infrastructure.

This tools costs $2495.

Products

Nagios Network Analyser

It gives the complete information of the network infrastructure to the admin with the potential threats on the network so that admin can take quick actions. It shares very detailed data about the network after in-depth network analysis.

This tools costs $1995.

Installation and Demo

Install Nagios build dependencies

- Sudo yum update –y
- Sudo yum intall gcc glibc glibc-common gd gd-devel make net-snmp openssl-devel xinetd unzip httpd –y
- sudo yum install php php-gd php-common –y

Creating Nagios user and groups

- sudo useradd nagios
- sudo groupadd nagcmd
- sudo usermod -a -G nagcmd nagios

Install Core Nagios

- cd ~
- curl -L -O
- https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.1.1.tar.gz
- tar xvf nagios-4.1.1.tar.gz
- cd nagios-4.1.1
- ./configure --with-command-group=nagcmd
- make all
- sudo make install
- sudo make install-commandmode
- sudo make install-init
- sudo make install-config
- sudo make install-webconf
- sudo usermod -G nagcmd apache

Install NRE

- cd ~
- curl -L –O http://downloads.sourceforge.net/project/nagios/nrpe-2.x/nrpe-2.15/nrpe-2.15.tar.gz
- tar xvf nrpe-2.15.tar.gz
- cd nrpe-2.15
- ./configure --enable-command-args --with-nagios-user=nagios --with-nagios-group=nagios --with-ssl=/usr/bin/openssl --with-ssl-lib=/usr/lib/x86_64-linux-gnu
- make all
- sudo make install
- sudo make install-xinetd
- sudo make install-daemon-config

Modify the only_from line by adding the private IP address of your server

- sudo vi /etc/xinetd.d/nrpe
- only_from = 127.0.0.1 10.132.224.168
- sudo service xinetd restart

Configuring Nagios

- uncomment this line #cfg_dir=/usr/local/nagios/etc/servers
- sudo vi /usr/local/nagios/etc/nagios.cfg
- Make directory
- sudo mkdir /usr/local/nagios/etc/servers
- Find the email directive, and replace its value with your own email address:
- sudo vi /usr/local/nagios/etc/objects/contacts.cfg
- email nagios@localhost ; <<***** CHANGE THIS TOYOUR EMAIL ADDRESS ******>>

Configuring check_nrpe Command

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

define command{

command_name check_nrpe

command_line $USER1$/check_nrpe -H $HOSTADDRESS$ -c

$ARG1$

}
```

Configuring Apache

Use htpasswd to create an admin user to access the Nagios web interface:

```
sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

Restart Apache:

```
sudo systemctl daemon-reload

sudo systemctl start nagios.service

sudo systemctl restart httpd.service

sudo chkconfig nagios on
```

Now go to the URL

http://server-ip/nagios

Add Hosts

```
mkdir /usr/local/nagios/etc/servers
vim /usr/local/nagios/etc/servers/docker2.cfg
define host {
use linux-server
host_name docker2
alias Docker server
address 192.168.90.62
max_check_attempts 5
check_period 24x7
notification_interval 30
notification_period 24x7
}
```

```
define service {
use generic-service
host_name docker2
service_description PING
check_command check_ping!100.0,20%!500.0,60%
}
define service {
use generic-service
host_name docker2
service_description SSH
check_command check_ssh
notifications_enabled 0
}
```

Thank you