

Advanced DevOps Lab Experiment 10

Aim: To perform Port, Service monitoring, Windows/Linux server monitoring using Nagios.

Theory:

Nagios is a comprehensive monitoring and alerting platform designed to keep track of IT infrastructure, networks, and applications. It provides real-time monitoring, alerting, and reporting capabilities to ensure the health and performance of critical systems.

Key Components of Nagios

- 1.Nagios Core:** The open-source foundation of the Nagios monitoring system. It provides the basic framework for monitoring and alerting.
- 2.Nagios XI:** A commercial version of Nagios that offers advanced features, a more user-friendly interface, and additional support options.
- 3.Nagios Log Server:** A tool for centralized log management, allowing you to view, analyze, and archive logs from various sources.
- 4.Nagios Network Analyzer:** Provides detailed insights into network traffic and bandwidth usage.
- 5.Nagios Fusion:** Centralizes monitoring data from multiple Nagios instances, providing a unified view of the entire networks.

How Nagios Works

- 1.Configuration:** Administrators define what to monitor and how to monitor it using configuration files.
- 2.Plugins:** Nagios uses plugins to gather information about the status of various services and hosts. These plugins can be custom scripts or pre-built ones.
- 3.Scheduling:** Nagios schedules regular checks of the defined services and hosts using the configured plugins.
- 4.Alerting:** If a check indicates a problem, Nagios triggers an alert. Alerts can be configured to escalate if not acknowledged within a certain timeframe.

5. Log Management: Centralizing and analyzing logs from various sources to detect issues and ensure compliance.

Implementation :

Prerequisites

- AWS Free Tier
- Nagios Server running on an Amazon Linux Machine

1. Confirm Nagios is Running on the Server

- `sudo systemctl status nagios`
- Proceed if you see that Nagios is active and running.

```
Things look okay - No serious problems were detected during the pre-flight check
[ec2-user@ip-172-31-42-50 nagios-plugins-2.3.3]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.6
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
   Active: active (running) since Mon 2024-10-07 16:28:45 UTC; 38s ago
     Docs: https://www.nagios.org/documentation
  Process: 69362 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
  Process: 69363 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (co
 Main PID: 69364 (nagios)
    Tasks: 6 (limit: 1112)
   Memory: 2.1M
      CPU: 22ms
  CGroup: /system.slice/nagios.service
          └─69364 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
          └─69365 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
          └─69366 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
          └─69367 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
          └─69368 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
          └─69369 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
```

2. Create an Ubuntu 20.04 Server EC2 Instance

- Name it linux-client.
- Use the same security group as the Nagios Host

EC2 > ... > Launch an instance

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name

[Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

3. Verify Nagios Process on the Server

Commands

- `ps -ef | grep nagios`

```
[ec2-user@ip-172-31-42-50 nagios-plugins-2.3.3]$ ps -ef | grep nagios
nagios      69364      1    0 16:28 ?        00:00:00 /usr/local/nagios/bin/nagios -d
nagios      69365     69364    0 16:28 ?        00:00:00 /usr/local/nagios/bin/nagios --v
nagios      69366     69364    0 16:28 ?        00:00:00 /usr/local/nagios/bin/nagios --v
nagios      69367     69364    0 16:28 ?        00:00:00 /usr/local/nagios/bin/nagios --v
nagios      69368     69364    0 16:28 ?        00:00:00 /usr/local/nagios/bin/nagios --v
nagios      69369     69364    0 16:28 ?        00:00:00 /usr/local/nagios/bin/nagios -d
ec2-user    70969     2909    0 16:55 pts/0    00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-42-50 nagios-plugins-2.3.3]$
```

4. Become Root User and Create

Directories

`sudo su`

- `mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts`

```
[ec2-user@ip-172-31-42-50 nagios-plugins-2.3.3]$ sudo su
mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-42-50 nagios-plugins-2.3.3]#
```

5. Copy Sample Configuration File

cp /usr/local/nagios/etc/objects/localhost.cfg

/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg

```
[root@ip-172-31-42-50 ec2-user]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/
[root@ip-172-31-42-50 ec2-user]# nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
[root@ip-172-31-42-50 ec2-user]#
```

6. Edit the Configuration File

sudo nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg

- Change hostname to linuxserver everywhere in the file.
- Change address to the public IP address of your linux-client.

```
#####
#
# HOST DEFINITION
#
#####
# Define a host for the local machine

define host {

    use                linux-server          ; Name of host template to use
                                           ; This host definition will inherit all variables that are defined
                                           ; in (or inherited by) the linux-server host template definition.

    host_name          linuxserver
    alias              linuxserver
    address             127.0.0.1
}

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo
```

7. Update Nagios Configuration

sudo nano /usr/local/nagios/etc/nagios.cfg

- Add the following line: cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/
- Change hostgroup_name under hostgroup to linux-servers1

```
#####
#
# HOST GROUP DEFINITION
#
#####

# Define an optional hostgroup for Linux machines

define hostgroup {

    hostgroup_name    linux-servers1      ; The name of the hostgroup
    alias              Linux Servers       ; Long name of the group
    members            localhost           ; Comma separated list of hosts that belong to this group
}

#####
```

8. Verify Configuration Files

`sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg`

```
[root@ip-172-31-42-50 ec2-user]# sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.4.6
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2020-04-28
License: GPL

Website: https://www.nagios.org
Reading configuration data...
    Read main config file okay...
    Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
    Checked 16 services.
    Checked 2 hosts.
    Checked 2 host groups.
    Checked 0 service groups.
    Checked 1 contacts.
    Checked 1 contact groups.
    Checked 24 commands.
    Checked 5 time periods.
    Checked 0 host escalations.
    Checked 0 service escalations.
```

```
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors:   0

Things look okay - No serious problems were detected during the pre-flight check
[root@ip-172-31-42-50 ec2-user]#
```

9. Restart Nagios Service

- `sudo systemctl restart nagios`

10. SSH into the Client Machine

- Use SSH or EC2 Instance Connect to access the linux-client.

11. Update Package Index and Install Required Packages

- `sudo apt update -y`
- `sudo apt install gcc -y`
- `sudo apt install -y nagios-nrpe-server nagios-plugins`

```
ubuntu@ip-172-31-33-27:~$ sudo apt update -y
sudo apt install gcc -y
sudo apt install -y nagios-nrpe-server nagios-plugins
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [9,418 kB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [53.9 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [4,791 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [9,418 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [4,791 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [2,911 kB]
```

12. Edit NRPE Configuration File Commands -

`sudo nano /etc/nagios/nrpe.cfg`

- Add your Nagios host IP address under `allowed_hosts`: `allowed_hosts=`

```
# supported.
#
# Note: The daemon only does rudimentary checking of the client's IP
# address. I would highly recommend adding entries in your /etc/hosts.allow
# file to allow only the specified host to connect to the port
# you are running this daemon on.
#
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd
allowed_hosts=127.0.0.1,3.81.151.142

# COMMAND ARGUMENT PROCESSING
# This option determines whether or not the NRPE daemon will allow clients
# to specify arguments to commands that are executed. This option only works
# if the daemon was configured with the --enable-command-args configure script
# option.
#
```

14. Check Nagios Dashboard

- Open your browser and navigate to <http://nagios>.
- Log in with nagiosadmin and the password you set earlier.
- You should see the new host linuxserver added.
- Click on Hosts to see the host details.
- Click on Services to see all services and ports being monitored

Nagios®

General
[Home](#)
[Documentation](#)

Current Status
[Tactical Overview](#)
[Map \(Legacy\)](#)
[Hosts](#)
[Services](#)
[Host Groups](#)
[Summary](#)
[Grid](#)
[Service Groups](#)
[Summary](#)
[Grid](#)
Problems
[Services \(Unhandled\)](#)
[Hosts \(Unhandled\)](#)
[Network Outages](#)
Quick Search:

Reports
[Availability](#)

Current Network Status
Last Updated: Mon Oct 7 18:26:34 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as nagiosadmin

Host Status Totals

Up	Down	Unreachable	Pending
2	0	0	0

All Problems: 0, All Types: 2

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
12	2	0	2	0

All Problems: 4, All Types: 16

View Service Status Detail For All Host Groups
View Status Overview For All Host Groups
View Status Summary For All Host Groups
View Status Grid For All Host Groups

Host Status Details For All Host Groups

Limit Results: 100

Host	Status	Last Check	Duration	Status Information
linuxserver	UP	10-07-2024 18:22:38	0d 0h 23m 18s	PING OK - Packet loss = 0%, RTA = 0.03 ms
localhost	UP	10-07-2024 18:23:07	0d 1h 57m 49s	PING OK - Packet loss = 0%, RTA = 0.03 ms

Results 1 - 2 of 2 Matching Hosts

Nagios®

General

[Home](#)
[Documentation](#)

Current Status

[Tactical Overview](#)
[Map \(Legacy\)](#)
[Hosts](#)
[Services](#)
[Host Groups](#)

Summary

Grid

[Service Groups](#)

Summary

Grid

[Problems](#)

Services

(Unhandled)

Hosts (Unhandled)

Network Outages

Quick Search:

Reports

[Availability](#)
[Trends \(Legacy\)](#)
[Alerts](#)

Host Information

Last Updated: Mon Oct 7 18:28:15 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as nagiosadmin

Host

linuxserver

(linuxserver)

Member of

No hostgroups

127.0.0.1

View Status Detail For This Host

View Alert History For This Host

View Trends For This Host

View Alert Histogram For This Host

View Availability Report For This Host

View Notifications For This Host

Host State Information

Host Status:

UP (for 0d 0h 24m 59s)

Status Information:

PING OK - Packet loss = 0%, RTA = 0.03 ms

Performance Data:

rtt=0.034000ms;3000.000000;5000.000000;0.000000
p1=0%;80;100;0
1/10 (HARD state)

Current Attempt:

1/10 (HARD state)

Last Check Time:

10-07-2024 18:27:38

Check Type:

ACTIVE

Check Latency / Duration:

0.000 / 4.160 seconds

Next Scheduled Active Check:

10-07-2024 18:32:38

Last State Change:

10-07-2024 18:03:16

Last Notification:

N/A (notification 0)

Is This Host Flapping?

NO (0.00% state change)

In Scheduled Downtime?

NO

Last Update:

10-07-2024 18:28:05 (0d 0h 0m 10s ago)

Active Checks:

ENABLED

Passive Checks:

ENABLED

Obsessing:

ENABLED

Nagios®

General

[Home](#)
[Documentation](#)

Current Status

[Tactical Overview](#)
[Map \(Legacy\)](#)
[Hosts](#)
[Services](#)
[Host Groups](#)

Summary

Grid

[Service Groups](#)

Summary

Grid

[Problems](#)

Services

(Unhandled)

Hosts (Unhandled)

Network Outages

Quick Search:

Reports

[Availability](#)
[Trends \(Legacy\)](#)
[Alerts](#)

Current Network Status

Last Updated: Mon Oct 7 18:33:39 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as nagiosadmin

View History For All Hosts
View Notifications For All Hosts
View Host Status Detail For All Hosts

Host Status Totals

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

Service Status Totals

Ok Warning Unknown Critical Pending
12 2 0 2 0
All Problems All Types
4 16

Service Status Details For All Hosts

Limit Results: 100

Host	Service	Status	Last Check	Duration	Attempt	Status Information
linuxserver	Current Load	OK	10-07-2024 18:28:53	0d 0h 26m 46s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	10-07-2024 18:29:31	0d 0h 26m 0s	1/4	USERS OK - 2 users currently logged in
	HTTP	WARNING	10-07-2024 18:33:08	0d 0h 25m 31s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 319 bytes in 0.000 second response time
	PING	OK	10-07-2024 18:30:46	0d 0h 27m 53s	1/4	PING OK - Packet loss = 0%, RTA = 0.03 ms
	Root Partition	OK	10-07-2024 18:31:23	0d 0h 27m 16s	1/4	DISK OK - free space: / 6000 MB (74.91% inode=98%)
	SSH	OK	10-07-2024 18:32:01	0d 0h 26m 38s	1/4	SSH OK - OpenSSH_8.7 (protocol 2.0)
	Swap Usage	CRITICAL	10-07-2024 18:30:38	0d 0h 23m 1s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size
localhost	Total Processes	OK	10-07-2024 18:33:16	0d 0h 25m 23s	1/4	PROCS OK, 37 processes with STATE = RSZDT
	Current Load	OK	10-07-2024 18:29:22	0d 2h 4m 17s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	10-07-2024 18:30:00	0d 2h 3m 39s	1/4	USERS OK - 2 users currently logged in
	HTTP	WARNING	10-07-2024 18:28:37	0d 2h 5m 2s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 319 bytes in 0.001 second response time
	PING	OK	10-07-2024 18:31:15	0d 2h 2m 24s	1/4	PING OK - Packet loss = 0%, RTA = 0.03 ms

Conclusion:

To perform port, service, and Windows/Linux server monitoring using Nagios, configure the necessary plugins and agents, define the monitoring parameters in the configuration files, and set up alerting mechanisms to ensure timely notifications of any issues. This comprehensive approach ensures robust monitoring and quick response to potential problems, maintaining the health and performance of your IT infrastructure.