Shravani Anil Patil D15A-38

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

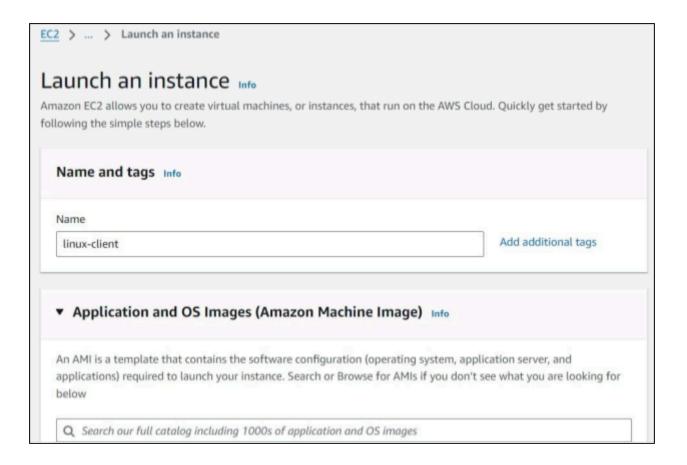
1. Confirm Nagios is Running on the Server

- sudo systemetl 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
               -69366 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
              -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



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 --n nagios 69366 69364 0 16:28 ? 00:00:00 /usr/local/nagios/bin/nagios --n nagios 69367 69364 0 16:28 ? 00:00:00 /usr/local/nagios/bin/nagios --n nagios 69368 69364 0 16:28 ? 00:00:00 /usr/local/nagios/bin/nagios --n nagios 69369 69364 0 16:28 ? 00:00:00 /usr/local/nagios/bin/nagios --n 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.

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

8. Verify Configuration Files

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 systemetl 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 InR
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports I
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe am
Get:6 http://security.ubuntu.com/ubuntu noble-security/main amd64 Pack
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Tr
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe am
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe am
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse
```

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

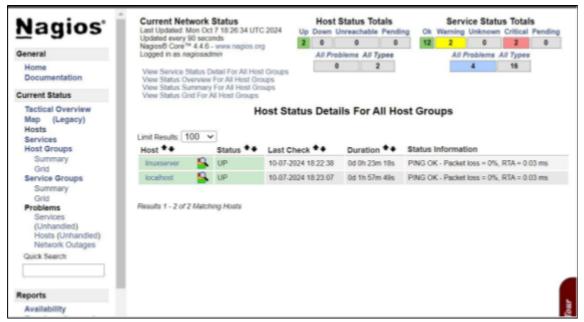
# if the daemon was configured with the --enable-command-args configure script

# 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







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.