Name: Hrishikesh Kumbhar

Div: D15A

Roll no: 32

Sub: Advanced DevOps

Experiment No: 10

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

Steps:

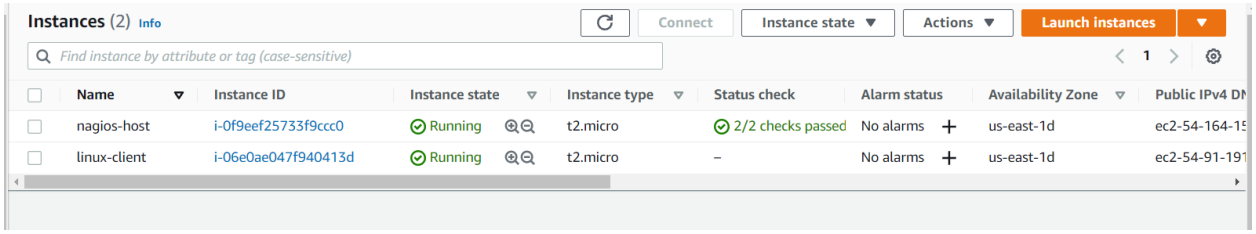
Prerequisites: AWS Free Tier, Nagios Server running on Amazon Linux Machine.

Step 1. To Confirm that Nagios is running on the server side, run this sudo systemctl status nagios on the "NAGIOS HOST".

```
[ec2-user@ip-172-31-29-80 nagios-plugins-2.0.3]$ sudo systemctl status nagios
● nagios.service - LSB: Starts and stops the Nagios monitoring server
   Loaded: loaded (/etc/rc.d/init.d/nagios; bad; vendor preset: disabled)
   Active: active (running) since Fri 2022-10-07 19:12:31 UTC; 42min ago
     Docs: man:systemd-sysv-generator(8)
  Process: 26732 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/nagios.service
           ├─26753 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
           ├─26755 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─26756 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─26757 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─26758 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           └─26759 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
```

You can proceed if you get this message.

Step 2. Before we begin, To monitor a Linux machine, create an Ubuntu 20.04 server EC2 Instance in AWS. Provide it with the same security group as the Nagios Host and name it 'linux-client' alongside the host.



For now, leave this machine as is, and go back to your nagios HOST machine.

Step 3. On the server, run this command

```
ps -ef | grep nagios
```

```
[ec2-user@ip-172-31-29-80 nagios-plugins-2.0.3]$ ps -ef | grep nagios
ec2-user  5675  3674  0 20:00 pts/0    00:00:00 grep --color=auto nagios
nagios   26753     1  0 19:12 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios   26755 26753  0 19:12 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   26756 26753  0 19:12 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   26757 26753  0 19:12 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   26758 26753  0 19:12 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   26759 26753  0 19:12 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
[ec2-user@ip-172-31-29-80 nagios-plugins-2.0.3]$
```

Step 4. Become a root user and create 2 folders

```
sudo su
mkdir /usr/local/nagios/etc/objects/monitorhosts
mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
```

Step 5. Copy the sample localhost.cfg file to linuxhost folder

```
cp /usr/local/nagios/etc/objects/localhost.cfg
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linux
server.cfg
```

Step 6. Open linuxserver.cfg using nano and make the following changes

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

Change the hostname to linuxserver (EVERYWHERE ON 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                 54.91.191.116
        }



##############################################################################
```

Change hostgroup_name under hostgroup to linux-servers1

```
# Define an optional hostgroup for Linux machines

define hostgroup{
        hostgroup_name  linux-servers ; The name of the hostgroup
        alias           linux-servers1 ; Long name of the group
        members         linuxserver    ; Comma separated list of hosts that belong to this group
        }
```

Everywhere else on the file, change the hostname to linuxserver instead of localhost.

Step 7. Open the Nagios Config file and add the following line

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

##Add this line

cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/

```
# directive as shown below:

#cfg_dir=/usr/local/nagios/etc/servers
#cfg_dir=/usr/local/nagios/etc/printers
#cfg_dir=/usr/local/nagios/etc/switches
#cfg_dir=/usr/local/nagios/etc/routers
cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/



# OBJECT CACHE FILE
# This option determines where object definitions are cached when
# Nagios starts/restarts.  The CGIs read object definitions from
# this cache file (rather than looking at the object config files
# directly) in order to prevent inconsistencies that can occur
# when the config files are modified after Nagios starts.

object_cache_file=/usr/local/nagios/var/objects.cache
```

Step 8. Verify the configuration files

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

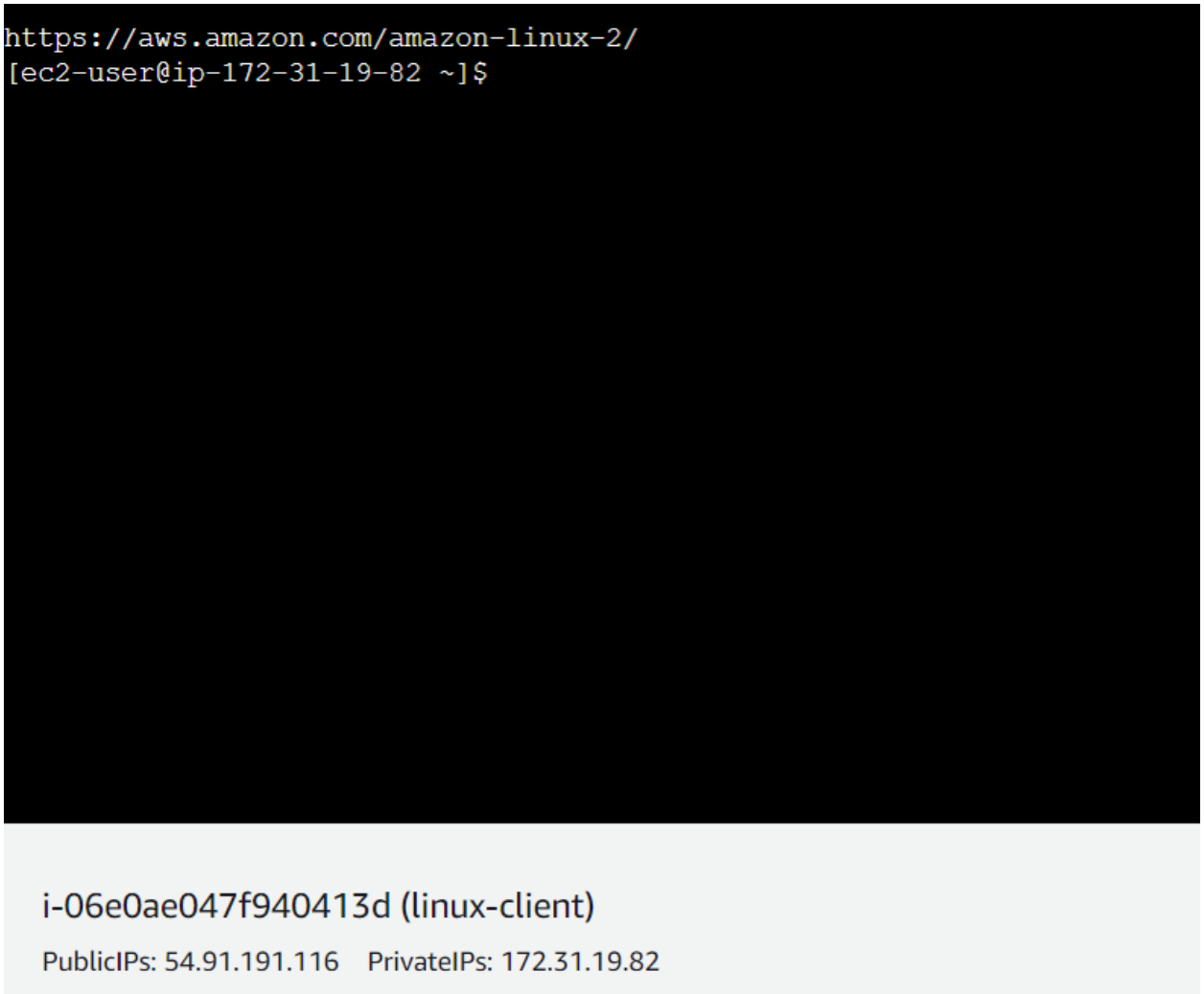
```
Running pre-flight check on configuration data...

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 for circular paths...
        Checked 2 hosts
        Checked 0 service dependencies
        Checked 0 host dependencies
        Checked 5 timeperiods
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-172-31-29-80 ec2-user]#
```

Step 9. Restart the nagios service

```
service nagios restart
```

```
[root@ip-172-31-29-80 ec2-user]# service nagios restart
Restarting nagios (via systemctl):                         [  OK  ]
[root@ip-172-31-29-80 ec2-user]#
```

Step 10. SSH into the machine or simply use the EC2 Instance Connect feature



Step 11. Make a package index update and install gcc, nagios-nrpe-server and the plugins.

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

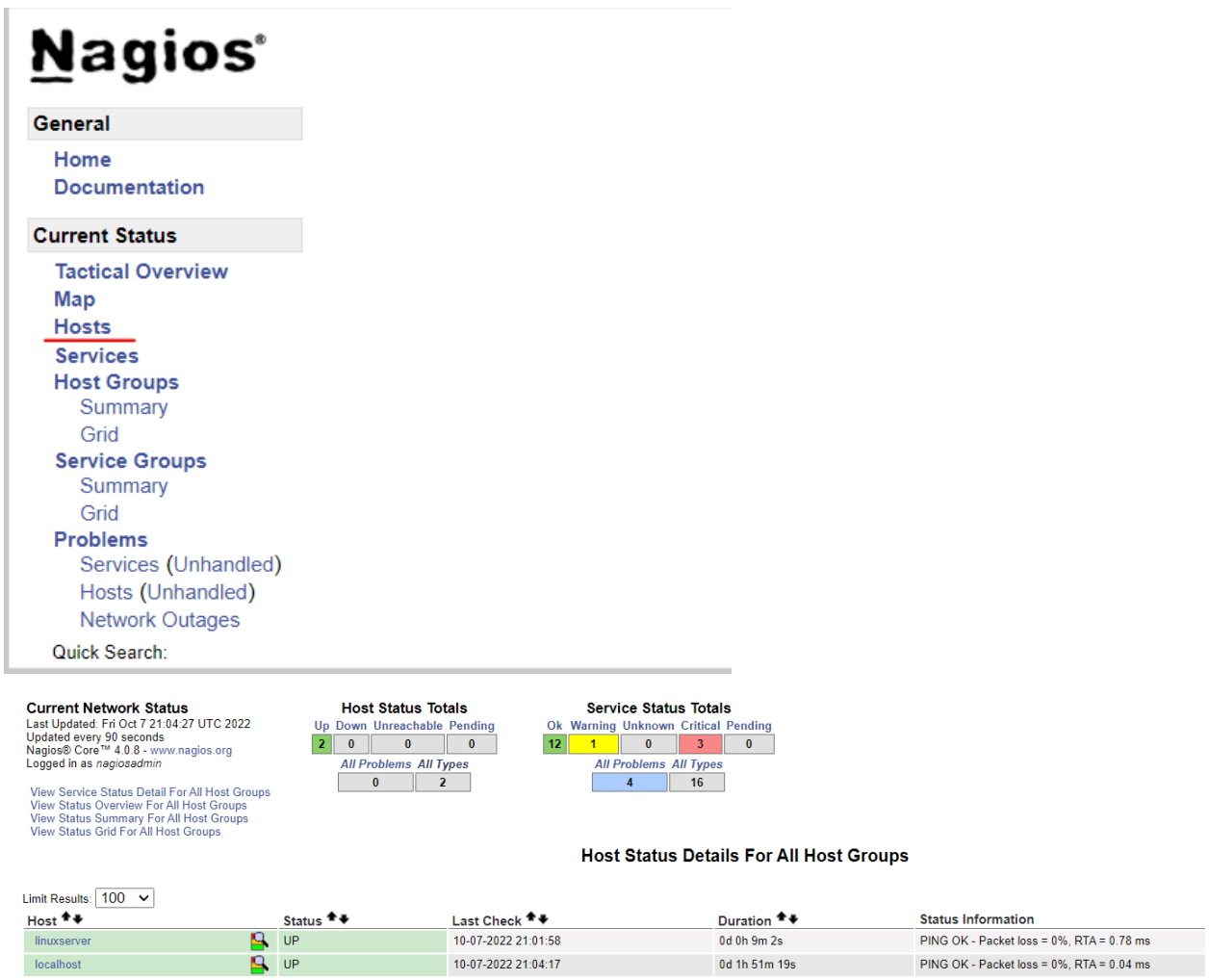
Step 12. Open nrpe.cfg file to make changes.
sudo nano /etc/nagios/nrpe.cfg
Under allowed_hosts, add your nagios host IP address like so

```
# 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,::1,54.164.157.242
```

Step 13. Restart the NRPE server

```
sudo systemctl restart nagios-nrpe-server
```

Step 14. Now, check your nagios dashboard and you'll see a new host being added.



Click on linuxserver to see the host details

Limit Results: 100

Host	Status	Last Check	Duration	Status Information
linuxserver	UP	10-07-2022 21:01:58	0d 0h 9m 2s	PING OK - Packet loss = 0%, RTA = 0.78 ms
localhost	UP	10-07-2022 21:04:17	0d 1h 51m 19s	PING OK - Packet loss = 0%, RTA = 0.04 ms

Results 1 - 2 of 2 Matching Hosts

Host State Information

Host Status:	UP (for 0d 0h 10m 10s)
Status Information:	PING OK - Packet loss = 0%, RTA = 0.69 ms
Performance Data:	rta=0.687000ms;3000.000000;5000.000000;0.000000 pl=0%;80;100;0
Current Attempt:	1/10 (HARD state)
Last Check Time:	10-07-2022 21:04:28
Check Type:	ACTIVE
Check Latency / Duration:	0.000 / 4.085 seconds
Next Scheduled Active Check:	10-07-2022 21:09:32
Last State Change:	10-07-2022 20:55:25
Last Notification:	N/A (notification 0)
Is This Host Flapping?	NO (0.00% state change)
In Scheduled Downtime?	NO
Last Update:	10-07-2022 21:05:26 (0d 0h 0m 9s ago)

Active Checks:	ENABLED
Passive Checks:	ENABLED
Obsessing:	ENABLED
Notifications:	ENABLED
Event Handler:	ENABLED
Flap Detection:	ENABLED

You can click Services to see all services and ports being monitored

Limit Results: 100

Host	Service	Status	Last Check	Duration	Attempt	Status Information
linuxserver	Current Load	OK	10-07-2022 21:05:13	0d 0h 26m 3s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	10-07-2022 21:05:51	0d 0h 25m 25s	1/4	USERS OK - 1 users currently logged in
	HTTP	CRITICAL	10-07-2022 21:04:28	0d 0h 24m 48s	4/4	connect to address 52.201.229.140 and port 80: Connection refused
	PING	OK	10-07-2022 21:05:06	0d 0h 6m 10s	1/4	PING OK - Packet loss = 0%, RTA = 1.03 ms
	Root Partition	OK	10-07-2022 21:02:43	0d 0h 23m 33s	1/4	DISK OK - free space: / 6338 MB (77% inode=98%):
	SSH	OK	10-07-2022 21:05:21	0d 0h 10m 55s	1/4	SSH OK - OpenSSH_8.9p1 Ubuntu-3 (protocol 2.0)
	Swap Usage	CRITICAL	10-07-2022 21:01:58	0d 0h 22m 18s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
	Total Processes	OK	10-07-2022 21:04:36	0d 0h 21m 40s	1/4	PROCS OK: 30 processes with STATE = RSZDT
localhost	Current Load	OK	10-07-2022 21:05:32	0d 1h 53m 8s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	10-07-2022 21:01:09	0d 1h 52m 30s	1/4	USERS OK - 1 users currently logged in
	HTTP	WARNING	10-07-2022 21:01:47	0d 1h 51m 53s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 3932 bytes in 0.001 second response time
	PING	OK	10-07-2022 21:02:24	0d 1h 51m 15s	1/4	PING OK - Packet loss = 0%, RTA = 0.04 ms
	Root Partition	OK	10-07-2022 21:03:02	0d 1h 50m 38s	1/4	DISK OK - free space: / 6338 MB (77% inode=98%):
	SSH	OK	10-07-2022 21:03:39	0d 1h 50m 0s	1/4	SSH OK - OpenSSH_7.4 (protocol 2.0)
	Swap Usage	CRITICAL	10-07-2022 21:04:17	0d 1h 49m 23s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
	Total Processes	OK	10-07-2022 21:01:30	0d 1h 48m 45s	1/4	PROCS OK: 30 processes with STATE = RSZDT

As you can see, we have our linuxserver up and running. It is showing critical status on HTTP due to permission errors and swap because there is no partition created.

In this case, we have monitored -

Servers: 1 linux server

Services: swap

Ports: 22, 80 (ssh, http)

Processes: User status, Current load, total processes, root partition, etc.

Conclusion:

Thus, we learned about service monitoring using Nagios and successfully monitored a Linux Server and monitored its different ports and services using Nagios and NRPE.