NAME : HIMESH PATHAI

ROLL NO. : 35

DIV. : D15A

SUBJECT : ADV. DEV-OPS

BATCH – B

CASE STUDY – 15

Case Study: Kubernetes Application with Basic Monitoring

Case Study: Kubernetes Application with Basic Monitoring

Objective: To Set up a Kubernetes-based application with basic monitoring.
Key Technologies: Kubernetes, Google Cloud Console, and Nagios.
Problem Statement: "Deploy a basic application (such as an Nginx server) on a Kubernetes cluster using Google Cloud Console, and monitor its status with Nagios."
Tasks:

- Deploy the Nginx server on a Kubernetes cluster using Google Cloud Console.
- Install and configure Nagios to monitor the Nginx pod's status.
- Ensure that Nagios can detect the Nginx pod's running status and notify when it is unavailable.

Note:

Due to the discontinuation of Kubernetes support in AWS Cloud9, this experiment uses Google Cloud Console, which offers a more robust platform for Kubernetes deployments and easier integration with monitoring tools like Nagios.

1. Introduction

In modern cloud computing, container orchestration is essential for ensuring scalability, availability, and fault tolerance. This case study examines the process of deploying an Nginx server on a Kubernetes cluster while using Nagios to monitor the application's health, focusing on the broader implications for cloud-native infrastructure.

2. Theoretical Overview

2.1 Containerization and Orchestration

Containerization packages an application and its dependencies into a lightweight, isolated environment, allowing consistent performance across various stages—development, testing, and production. Kubernetes, an open-source orchestration platform, automates the deployment and scaling of containerized applications, ensuring optimal resource use and high reliability.

Key Concepts:

- Containers: Isolated environments for applications.
- Orchestration: Managing multiple containers to ensure smooth operation across distributed systems.
- Microservices Architecture: An approach where applications are divided into loosely coupled services, improving modularity and scalability.

2.2 Monitoring in Distributed Systems

Monitoring plays a vital role in maintaining application performance. In distributed environments like Kubernetes, traditional monitoring methods may be insufficient due to the dynamic nature of containerized applications. Nagios, a popular open-source tool, is capable of tracking application health, availability, and performance through customizable checks and alerts.

Key Concepts:

- Health Checks: Regular assessments of application and service status.
- Alerts: Notifications triggered by specific conditions indicating potential issues.
- Service-Level Objectives (SLOs): Metrics that define expected performance and reliability levels.

3. Methodology

3.1 Environment Setup

1. Cloud Provider Choice:
   - Due to the discontinuation of AWS Cloud9 for Kubernetes, Google Cloud Console was chosen for deploying and managing the Kubernetes cluster.
2. Kubernetes Cluster Creation:
   - A Kubernetes cluster was created in Google Cloud Console, following recommended practices for setup and management.

3.2 Application Deployment

1. Nginx Deployment:
   - The Nginx server was deployed using Kubernetes deployment manifests, which specify the application's desired state, including replicas, container images, and service configuration.
2. Service Exposure:
   - The Nginx server was exposed via a LoadBalancer service, allowing external access to the application.

3.3 Monitoring Setup

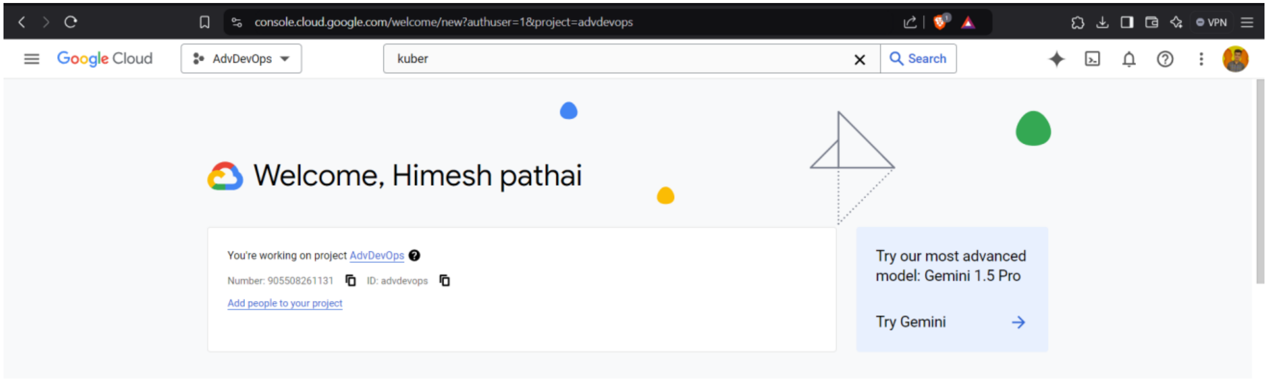
1. Nagios Installation:
   - Nagios was installed on a separate virtual machine, adhering to its installation and configuration documentation.
2. Monitoring Configuration:
   - Nagios was set up to monitor the Nginx server by defining checks that assess the application's availability and performance. Custom commands were used to run HTTPbased health checks**. 3. Alerting Mechanism:**
   - Nagios was configured to send email alerts when the Nginx server became unreachable, ensuring timely notifications for the operations team.

4.Steps performed:

Step 1:Make a account on Google cloud console:

https://console.cloud.google.com/

After making account, Screen will be as shown below:



console.cloud.google.com/welcome/new?authuser=1&project=advdevops

Welcome, Himesh pathai

You're working on project AdvDevOps

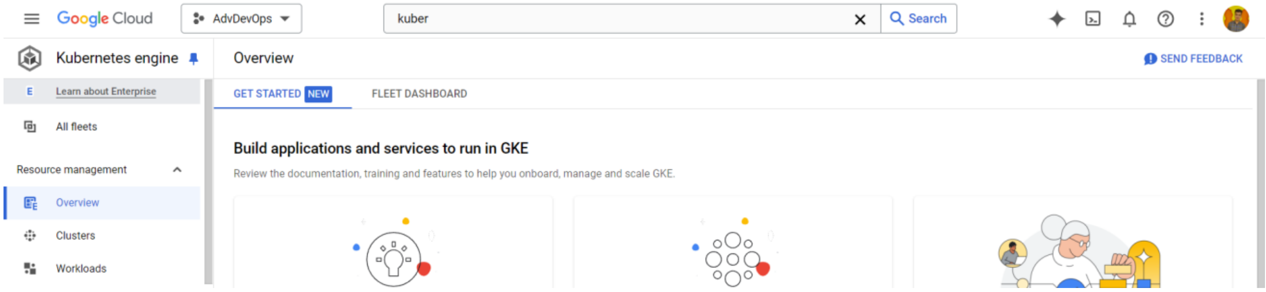
Number: 905508261131 ID: advdevops

Add people to your project

Try our most advanced model: Gemini 1.5 Pro

Try Gemini

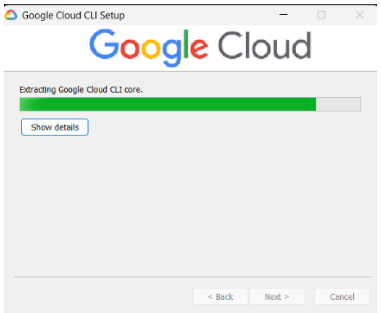
Step2:Now you need to make a new project:



Build applications and services to run in GKE

Review the documentation, training and features to help you onboard, manage and scale GKE.

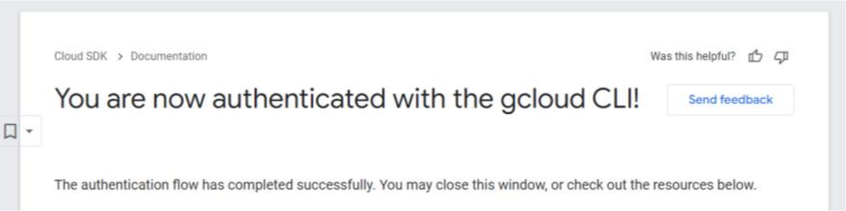
Step 3:Install Google cloud sdk
https://dl.google.com/dl/cloudsdk/channels/rapid/GoogleCloudSDKInstaller.exe From this link



Extracting Google Cloud CLI core.

Step 4:

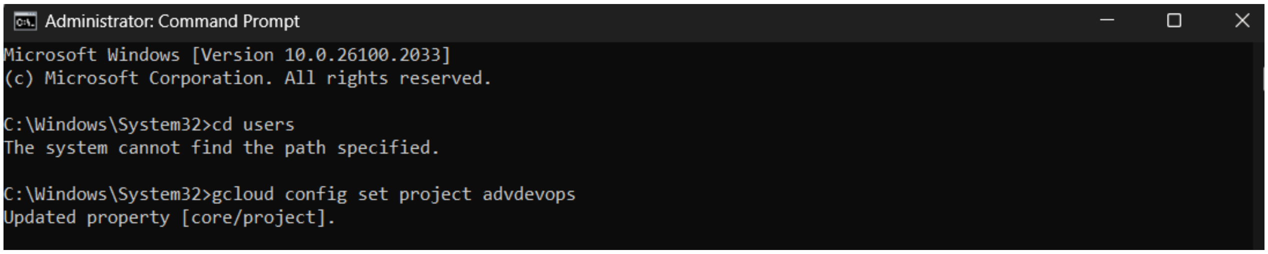
Authenticate your account



You are now authenticated with the gcloud CLI!

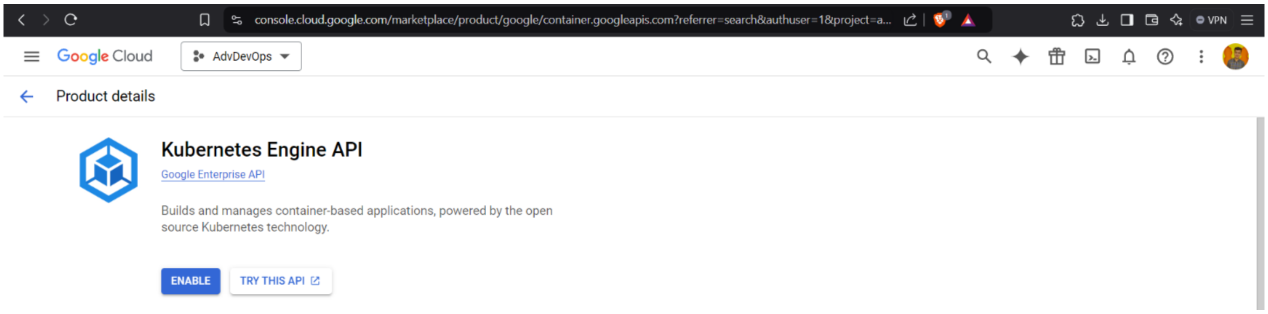
The authentication flow has completed successfully. You may close this window, or check out the resources below.

gcloud auth login
Set your GCP project
gcloud config set project <PROJECT_ID>
gcloud config set project advdevops



```
Microsoft Windows [Version 10.0.26100.2033]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>cd users
The system cannot find the path specified.

C:\Windows\System32>gcloud config set project advdevops
Updated property [core/project].
```

Step 5:Enable Kubernetes API:



console.cloud.google.com/marketplace/product/google/container.googleapis.com?referrer=search&authuser=1&project=a...

Product details

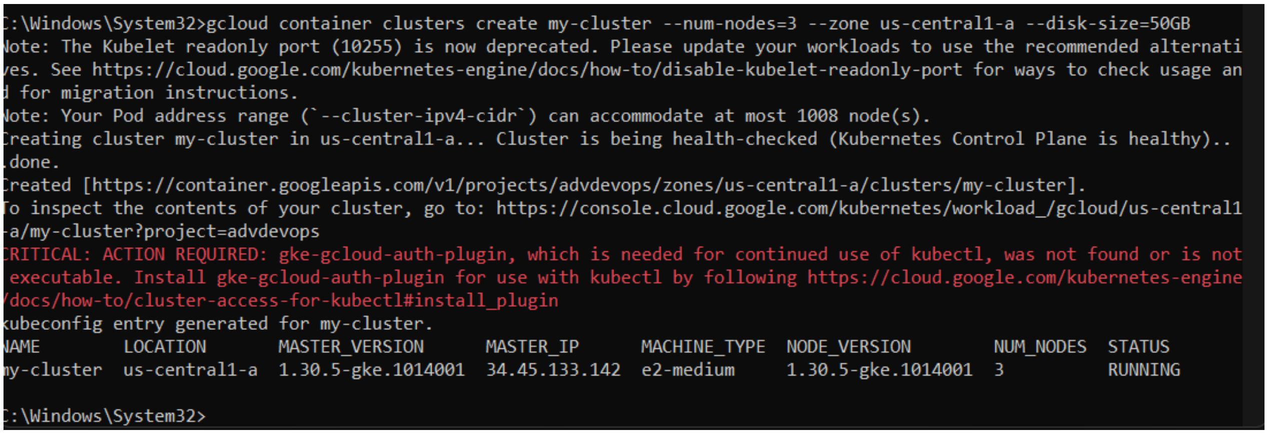
Kubernetes Engine API

Google Enterprise API

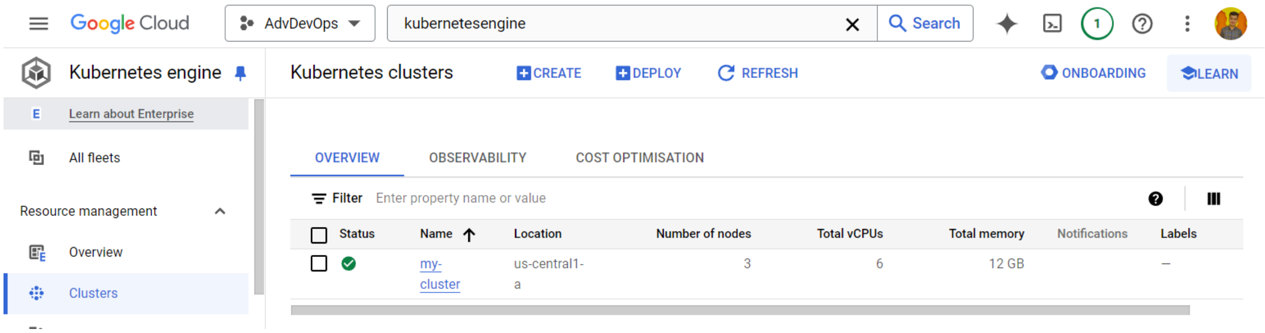
Builds and manages container-based applications, powered by the open source Kubernetes technology.

Step 6:use gcloud to create a GKE cluster from your terminal

gcloud container clusters create my-cluster --num-nodes=3 --zone us-central1-a



```
:\Windows\System32>gcloud container clusters create my-cluster --num-nodes=3 --zone us-central1-a --disk-size=50GB
Note: The Kubelet readonly port (10255) is now deprecated. Please update your workloads to use the recommended alternati
ves. See https://cloud.google.com/kubernetes-engine/docs/how-to/disable-kubelet-readonly-port for ways to check usage an
d for migration instructions.
Note: Your Pod address range (`--cluster-ipv4-cidr`) can accommodate at most 1008 node(s).
Creating cluster my-cluster in us-central1-a... Cluster is being health-checked (Kubernetes Control Plane is healthy)..
.done.
Created [https://container.googleapis.com/v1/projects/advdevops/zones/us-central1-a/clusters/my-cluster].
To inspect the contents of your cluster, go to: https://console.cloud.google.com/kubernetes/workload_/gcloud/us-central1
-a/my-cluster?project=advdevops
CRITICAL: ACTION REQUIRED: gke-gcloud-auth-plugin, which is needed for continued use of kubectl, was not found or is not
 executable. Install gke-gcloud-auth-plugin for use with kubectl by following https://cloud.google.com/kubernetes-engine
/docs/how-to/cluster-access-for-kubectl#install_plugin
kubeconfig entry generated for my-cluster.
NAME        LOCATION       MASTER_VERSION      MASTER_IP      MACHINE_TYPE  NODE_VERSION        NUM_NODES  STATUS
my-cluster  us-central1-a  1.30.5-gke.1014001  34.45.133.142  e2-medium     1.30.5-gke.1014001  3          RUNNING

C:\Windows\System32>
```



Kubernetes clusters

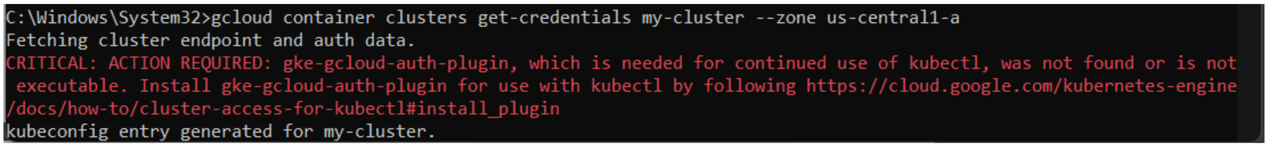
□	Status	Name ↑	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
□	✓	my-cluster	us-central1-a	3	6	12 GB		–

Step 6:

gcloud container clusters get-credentials my-cluster --zone us-central1-a

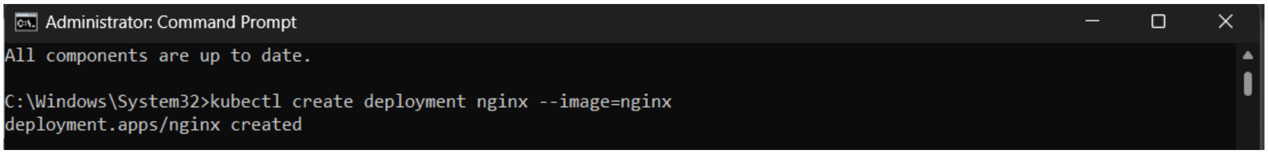
Connect to GKE Cluster

Get cluster credentials to interact with the Kubernetes cluster



```
C:\Windows\System32>gcloud container clusters get-credentials my-cluster --zone us-central1-a
Fetching cluster endpoint and auth data.
CRITICAL: ACTION REQUIRED: gke-gcloud-auth-plugin, which is needed for continued use of kubectl, was not found or is not
 executable. Install gke-gcloud-auth-plugin for use with kubectl by following https://cloud.google.com/kubernetes-engine
/docs/how-to/cluster-access-for-kubectl#install_plugin
kubeconfig entry generated for my-cluster.
```

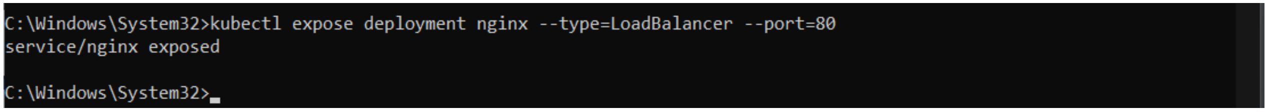
Step 7:
Create Nginx Deployment
Use kubectl to deploy an Nginx server : kubectl create deployment nginx --image=nginx



```
All components are up to date.

C:\Windows\System32>kubectl create deployment nginx --image=nginx
deployment.apps/nginx created
```

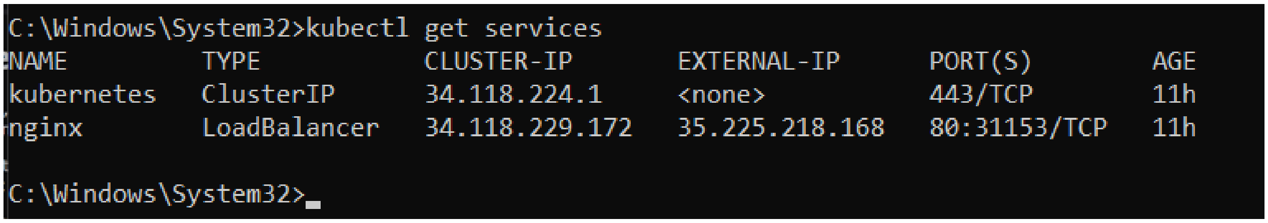
Step 8:Expose the Nginx deployment as a service kubectl expose deployment nginx --type=LoadBalancer --port=80

This creates a load balancer that allows you to access the Nginx application externally.



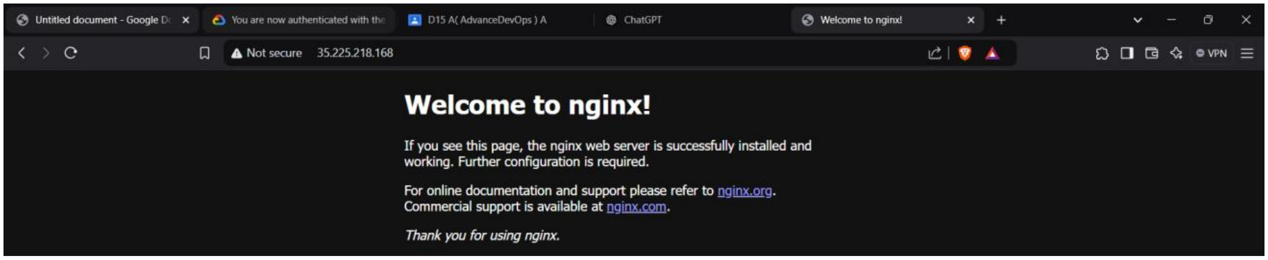
```
C:\Windows\System32>kubectl expose deployment nginx --type=LoadBalancer --port=80
service/nginx exposed

C:\Windows\System32>_
```

Step 9:
To get the external IP address of the service: kubectl get services



```
C:\Windows\System32>kubectl get services
NAME         TYPE           CLUSTER-IP       EXTERNAL-IP      PORT(S)        AGE
kubernetes   ClusterIP      34.118.224.1     <none>           443/TCP        11h
nginx        LoadBalancer   34.118.229.172   35.225.218.168   80:31153/TCP   11h

C:\Windows\System32>_
```

Once the external IP is available, you can access the Nginx server at http://34.135.244.240



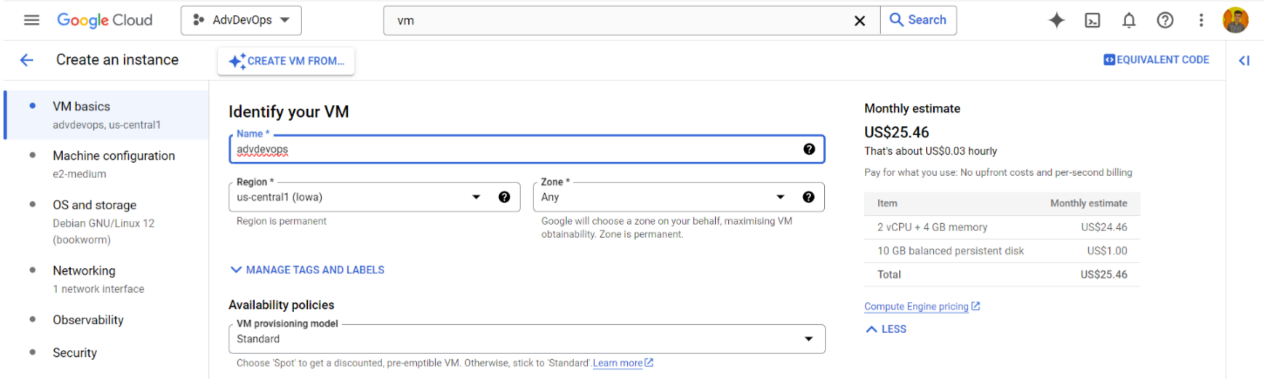
Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Step10: Create a VM Instance:



Identify your VM

Name * : advdevops

Region * : us-central1 (Iowa)
Region is permanent

Zone * : Any
Google will choose a zone on your behalf, maximising VM obtainability. Zone is permanent.

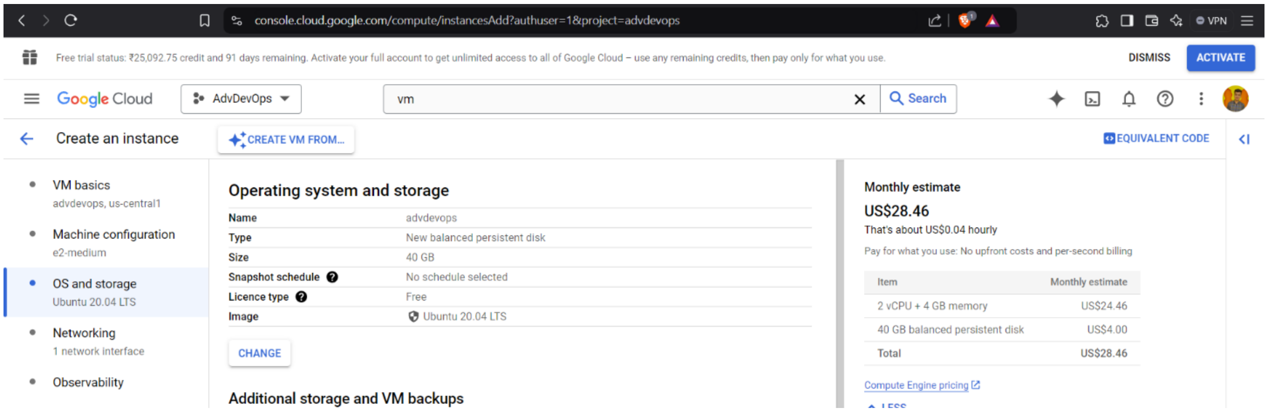
Availability policies

VM provisioning model : Standard
Choose 'Spot' to get a discounted, pre-emptible VM. Otherwise, stick to 'Standard'. Learn more

Monthly estimate
US$25.46
That's about US$0.03 hourly
Pay for what you use: No upfront costs and per-second billing

Item	Monthly estimate
2 vCPU + 4 GB memory	US$24.46
10 GB balanced persistent disk	US$1.00
Total	US$25.46

Do these configurations;



console.cloud.google.com/compute/instancesAdd?authuser=1&project=advdevops
Free trial status: ₹25,092.75 credit and 91 days remaining. Activate your full account to get unlimited access to all of Google Cloud – use any remaining credits, then pay only for what you use.
DISMISS
ACTIVATE
Google Cloud
AdvDevOps
vm
Search
Create an instance
CREATE VM FROM...
EQUIVALENT CODE
VM basics
advdevops, us-central1
Machine configuration
e2-medium
OS and storage
Ubuntu 20.04 LTS
Networking
1 network interface
Observability
CHANGE
Compute Engine pricing

Operating system and storage

Name	advdevops
Type	New balanced persistent disk
Size	40 GB
Snapshot schedule	No schedule selected
Licence type	Free
Image	Ubuntu 20.04 LTS

Additional storage and VM backups

Monthly estimate

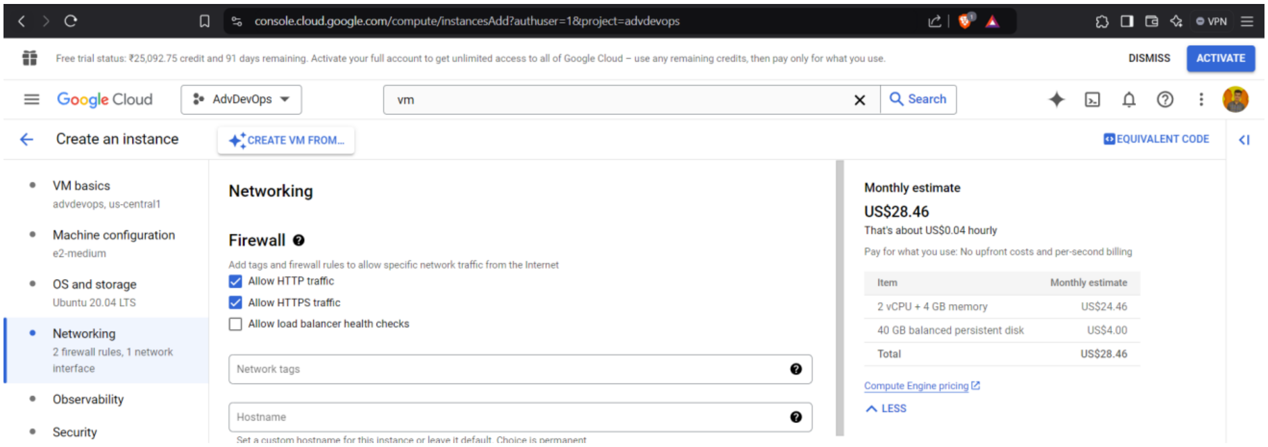
US$28.46

That's about US$0.04 hourly

Pay for what you use: No upfront costs and per-second billing

Item	Monthly estimate
2 vCPU + 4 GB memory	US$24.46
40 GB balanced persistent disk	US$4.00
Total	US$28.46

Allow HTTP
Allow HTTPS



console.cloud.google.com/compute/instancesAdd?authuser=1&project=advdevops
Free trial status: ₹25,092.75 credit and 91 days remaining. Activate your full account to get unlimited access to all of Google Cloud – use any remaining credits, then pay only for what you use.
DISMISS
ACTIVATE
Google Cloud
AdvDevOps
vm
Search
Create an instance
CREATE VM FROM...
EQUIVALENT CODE
VM basics
advdevops, us-central1
Machine configuration
e2-medium
OS and storage
Ubuntu 20.04 LTS
Networking
2 firewall rules, 1 network interface
Observability
Security
Network tags
Hostname
Compute Engine pricing
LESS

Networking

Firewall

Add tags and firewall rules to allow specific network traffic from the Internet

☑ Allow HTTP traffic

☑ Allow HTTPS traffic

□ Allow load balancer health checks

Set a custom hostname for this instance or leave it default. Choice is permanent

Monthly estimate

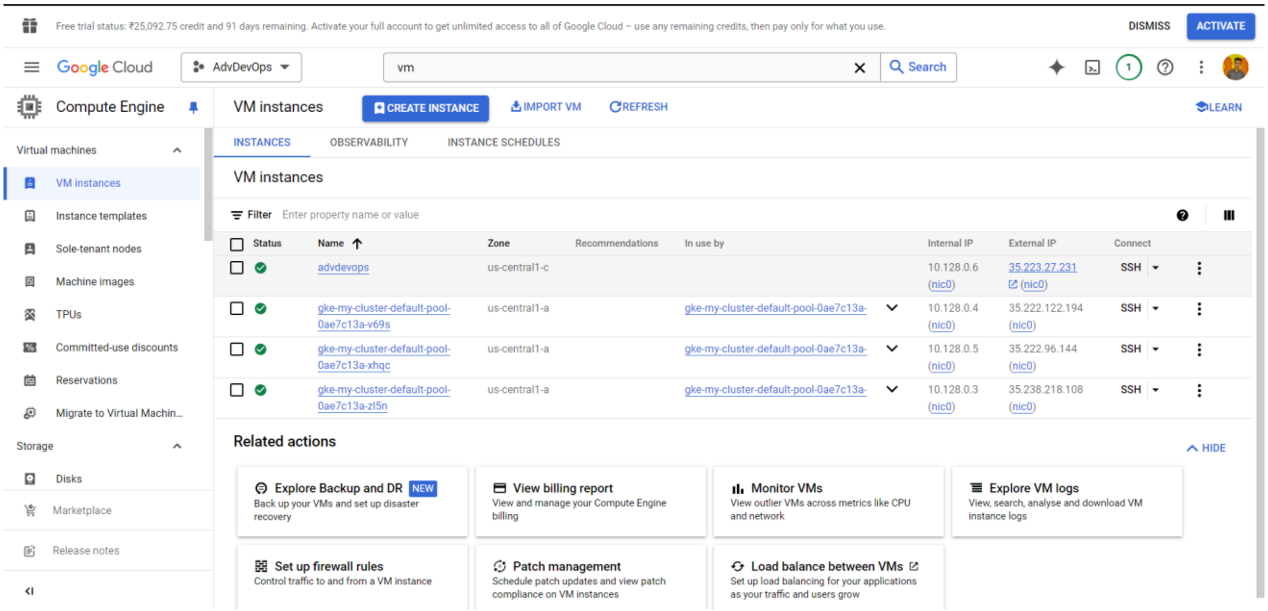
US$28.46

That's about US$0.04 hourly

Pay for what you use: No upfront costs and per-second billing

Item	Monthly estimate
2 vCPU + 4 GB memory	US$24.46
40 GB balanced persistent disk	US$4.00
Total	US$28.46

Your instance will get created



VM instances

Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
✓	advdevops	us-central1-c			10.128.0.6 (nic0)	35.223.27.231 (nic0)	SSH
✓	gke-my-cluster-default-pool-0ae7c13a-v69s	us-central1-a		gke-my-cluster-default-pool-0ae7c13a-	10.128.0.4 (nic0)	35.222.122.194 (nic0)	SSH
✓	gke-my-cluster-default-pool-0ae7c13a-xhqc	us-central1-a		gke-my-cluster-default-pool-0ae7c13a-	10.128.0.5 (nic0)	35.222.96.144 (nic0)	SSH
✓	gke-my-cluster-default-pool-0ae7c13a-zl5n	us-central1-a		gke-my-cluster-default-pool-0ae7c13a-	10.128.0.3 (nic0)	35.238.218.108 (nic0)	SSH

Related actions

Explore Backup and DR NEW
Back up your VMs and set up disaster recovery

View billing report
View and manage your Compute Engine billing

Monitor VMs
View outlier VMs across metrics like CPU and network

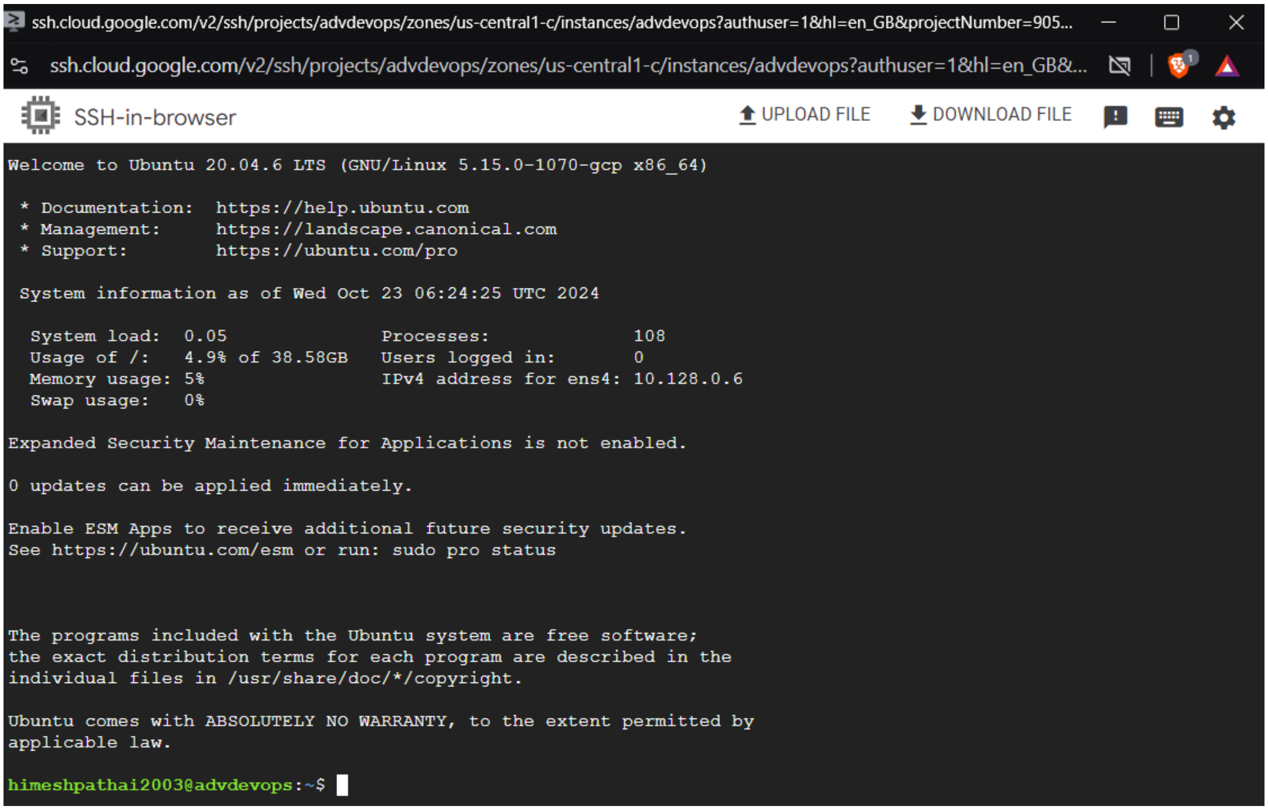
Explore VM logs
View, search, analyse and download VM instance logs

Set up firewall rules
Control traffic to and from a VM instance

Patch management
Schedule patch updates and view patch compliance on VM instances

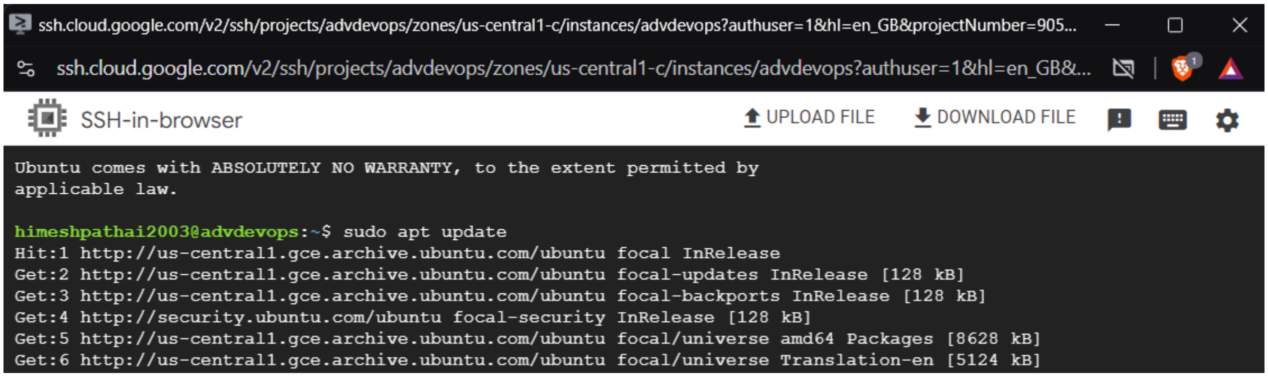
Load balance between VMs
Set up load balancing for your applications as your traffic and users grow

Step 11:Connect through SSH into the VM and install the necessary dependencies:


ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&projectNumber=905...
ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&...
SSH-in-browser
UPLOAD FILE
DOWNLOAD FILE

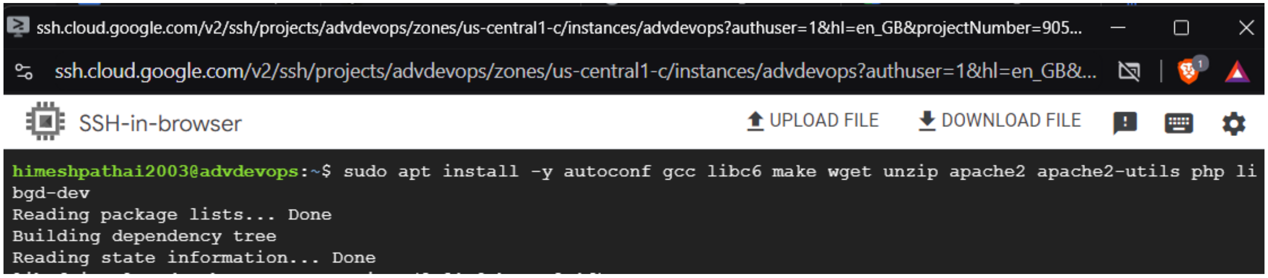
```
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-1070-gcp x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Wed Oct 23 06:24:25 UTC 2024

  System load:  0.05              Processes:             108
  Usage of /:   4.9% of 38.58GB   Users logged in:       0
  Memory usage: 5%                IPv4 address for ens4: 10.128.0.6
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

himeshpathai2003@advdevops:~$
```

sudo apt update


ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&projectNumber=905...
ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&...
SSH-in-browser
UPLOAD FILE
DOWNLOAD FILE

```
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

himeshpathai2003@advdevops:~$ sudo apt update
Hit:1 http://us-central1.gce.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://us-central1.gce.archive.ubuntu.com/ubuntu focal-updates InRelease [128 kB]
Get:3 http://us-central1.gce.archive.ubuntu.com/ubuntu focal-backports InRelease [128 kB]
Get:4 http://security.ubuntu.com/ubuntu focal-security InRelease [128 kB]
Get:5 http://us-central1.gce.archive.ubuntu.com/ubuntu focal/universe amd64 Packages [8628 kB]
Get:6 http://us-central1.gce.archive.ubuntu.com/ubuntu focal/universe Translation-en [5124 kB]
```

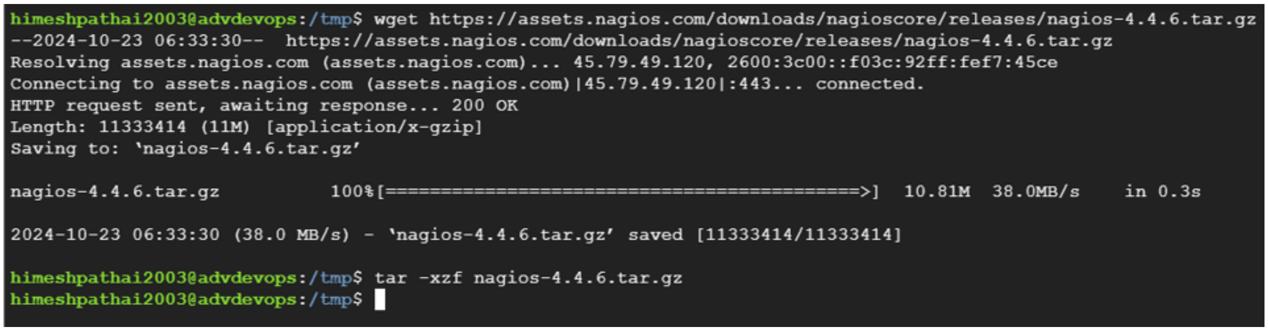
Step 12:Install dependency: sudo apt install -y autoconf gcc libc6 make wget unzip apache2 apache2-utils php libgd-dev


ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&projectNumber=905...
ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&...
SSH-in-browser
UPLOAD FILE
DOWNLOAD FILE

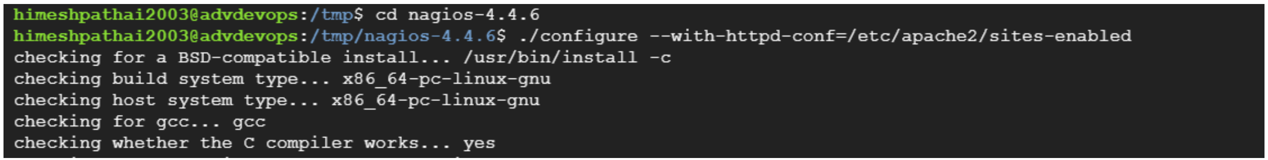
```
himeshpathai2003@advdevops:~$ sudo apt install -y autoconf gcc libc6 make wget unzip apache2 apache2-utils php li
bgd-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Step 13:Download and install Nagios

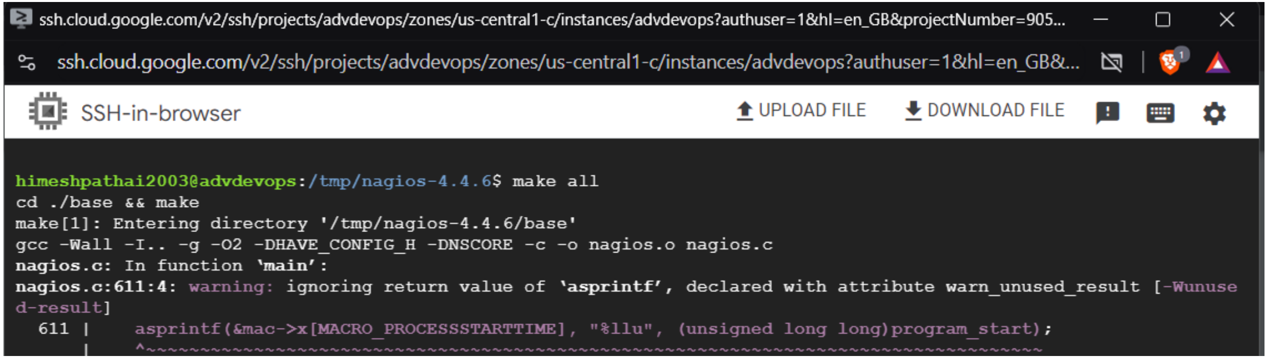
- cd /tmp
- wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
- tar -xzf nagios-4.4.6.tar.gz
- cd nagios-4.4.6
- ./configure --with-httpd-conf=/etc/apache2/sites-enabled
- make all



```
himeshpathai2003@advdevops:/tmp$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
--2024-10-23 06:33:30--  https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
Resolving assets.nagios.com (assets.nagios.com)... 45.79.49.120, 2600:3c00::f03c:92ff:fef7:45ce
Connecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11333414 (11M) [application/x-gzip]
Saving to: ‘nagios-4.4.6.tar.gz’

nagios-4.4.6.tar.gz          100%[===========================================>]  10.81M  38.0MB/s    in 0.3s

2024-10-23 06:33:30 (38.0 MB/s) - ‘nagios-4.4.6.tar.gz’ saved [11333414/11333414]

himeshpathai2003@advdevops:/tmp$ tar -xzf nagios-4.4.6.tar.gz
himeshpathai2003@advdevops:/tmp$ 
```



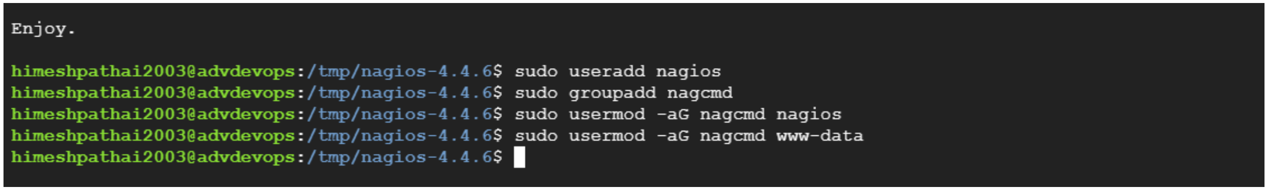
```
himeshpathai2003@advdevops:/tmp$ cd nagios-4.4.6
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ ./configure --with-httpd-conf=/etc/apache2/sites-enabled
checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking for gcc... gcc
checking whether the C compiler works... yes
```



ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&projectNumber=905...
ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&...
SSH-in-browser
UPLOAD FILE
DOWNLOAD FILE

```
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ make all
cd ./base && make
make[1]: Entering directory '/tmp/nagios-4.4.6/base'
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o nagios.c
nagios.c: In function ‘main’:
nagios.c:611:4: warning: ignoring return value of ‘asprintf’, declared with attribute warn_unused_result [-Wunuse
d-result]
  611 |    asprintf(&mac->x[MACRO_PROCESSSTARTTIME], "%llu", (unsigned long long)program_start);
      |    ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

Step 14:Add groups: sudo useradd nagios sudo groupadd nagcmd sudo usermod -aG nagcmd nagios sudo usermod -aG nagcmd www-data



```
Enjoy.

himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ sudo useradd nagios
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ sudo groupadd nagcmd
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ sudo usermod -aG nagcmd nagios
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ sudo usermod -aG nagcmd www-data
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ 
```

Step 15:Perform rest steps for installing nagios:

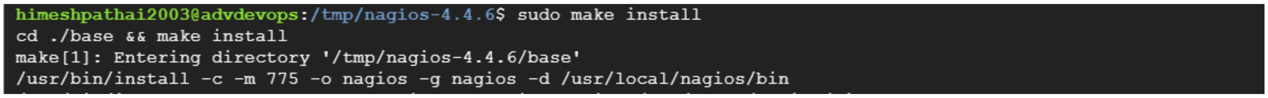
sudo make install

sudo make install-init

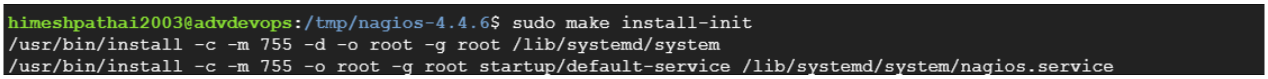
sudo make install-commandmode

sudo make install-config

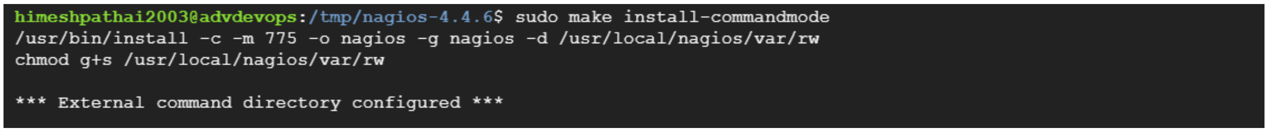
sudo make install-webconf



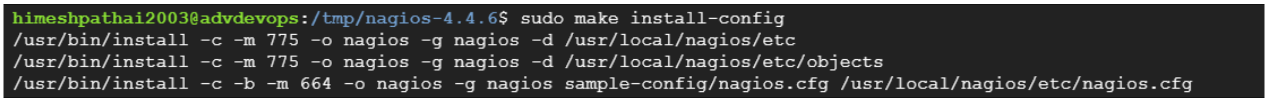
```
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ sudo make install
cd ./base && make install
make[1]: Entering directory '/tmp/nagios-4.4.6/base'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/bin
```



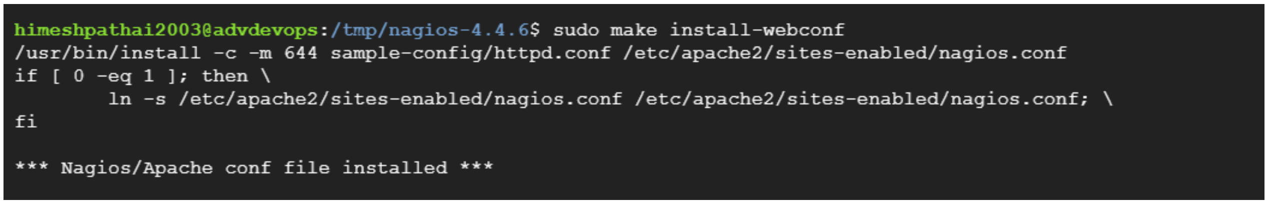
```
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ sudo make install-init
/usr/bin/install -c -m 755 -d -o root -g root /lib/systemd/system
/usr/bin/install -c -m 755 -o root -g root startup/default-service /lib/systemd/system/nagios.service
```



```
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ sudo make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw

*** External command directory configured ***
```



```
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ sudo make install-config
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc/objects
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg /usr/local/nagios/etc/nagios.cfg
```

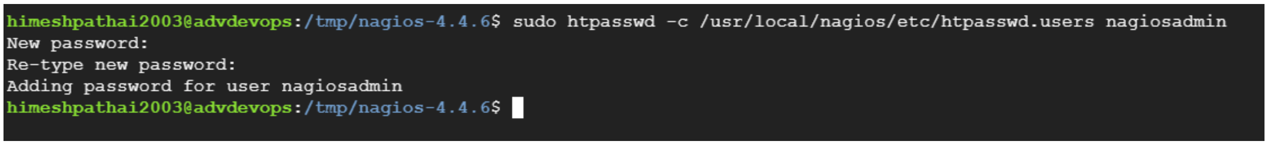


```
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ sudo make install-webconf
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/apache2/sites-enabled/nagios.conf
if [ 0 -eq 1 ]; then \
        ln -s /etc/apache2/sites-enabled/nagios.conf /etc/apache2/sites-enabled/nagios.conf; \
fi

*** Nagios/Apache conf file installed ***
```

Step 16:

Create a Nagios admin user

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



```
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ █
```

Step 17:

Install the Nagios plugins:

cd /tmp

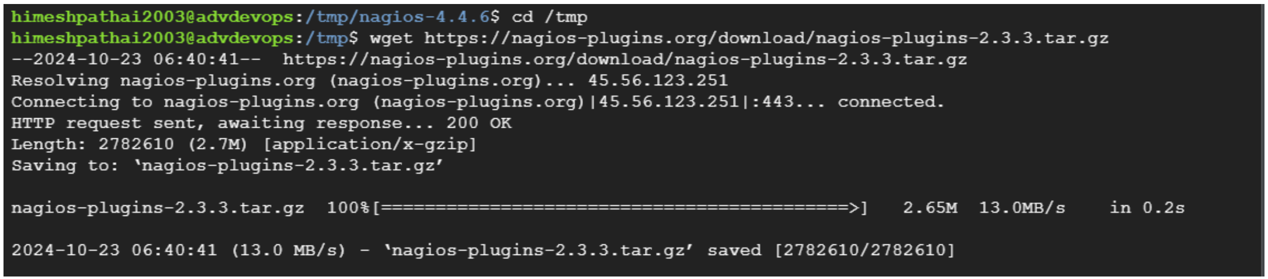
wget https://nagios-plugins.org/download/nagios-plugins2.3.3.tar.gz

tar -xzf nagios-plugins-2.3.3.tar.gz

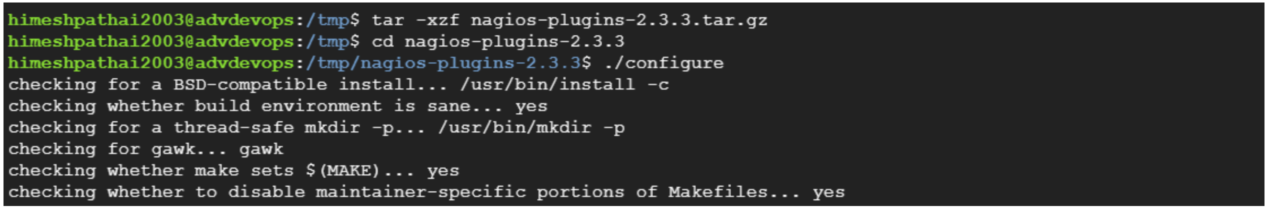
cd nagios-plugins-2.3.3

./configure

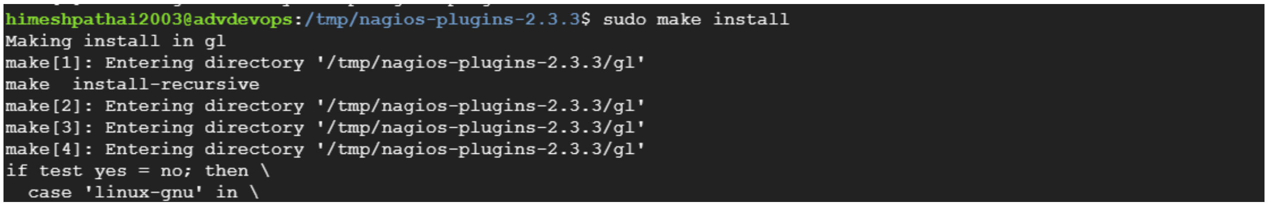
make sudo make install



```
himeshpathai2003@advdevops:/tmp/nagios-4.4.6$ cd /tmp
himeshpathai2003@advdevops:/tmp$ wget https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
--2024-10-23 06:40:41--  https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org)... 45.56.123.251
Connecting to nagios-plugins.org (nagios-plugins.org)|45.56.123.251|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2782610 (2.7M) [application/x-gzip]
Saving to: ‘nagios-plugins-2.3.3.tar.gz’

nagios-plugins-2.3.3.tar.gz  100%[==========================================>]   2.65M  13.0MB/s    in 0.2s

2024-10-23 06:40:41 (13.0 MB/s) - ‘nagios-plugins-2.3.3.tar.gz’ saved [2782610/2782610]
```

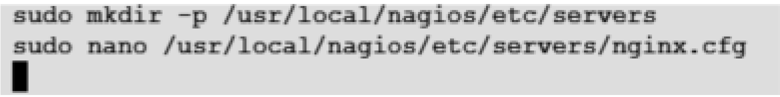


```
himeshpathai2003@advdevops:/tmp$ tar -xzf nagios-plugins-2.3.3.tar.gz
himeshpathai2003@advdevops:/tmp$ cd nagios-plugins-2.3.3
himeshpathai2003@advdevops:/tmp/nagios-plugins-2.3.3$ ./configure
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /usr/bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether to disable maintainer-specific portions of Makefiles... yes
```

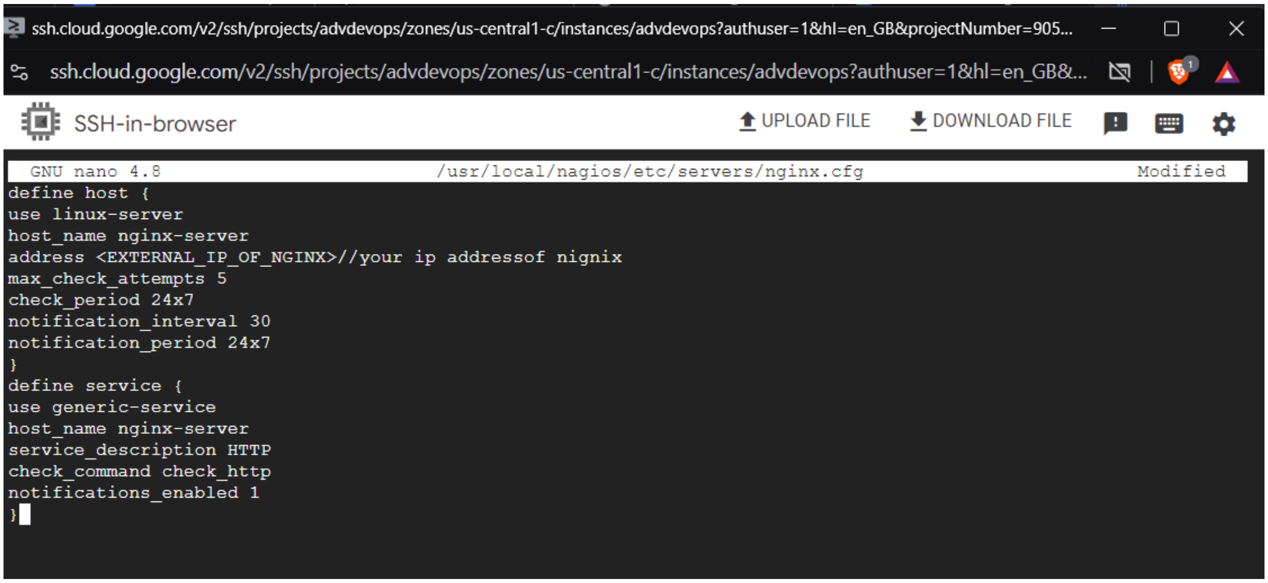


```
himeshpathai2003@advdevops:/tmp/nagios-plugins-2.3.3$ sudo make install
Making install in gl
make[1]: Entering directory '/tmp/nagios-plugins-2.3.3/gl'
make  install-recursive
make[2]: Entering directory '/tmp/nagios-plugins-2.3.3/gl'
make[3]: Entering directory '/tmp/nagios-plugins-2.3.3/gl'
make[4]: Entering directory '/tmp/nagios-plugins-2.3.3/gl'
if test yes = no; then \
  case 'linux-gnu' in \
```

Step 19:Configure and make new file:



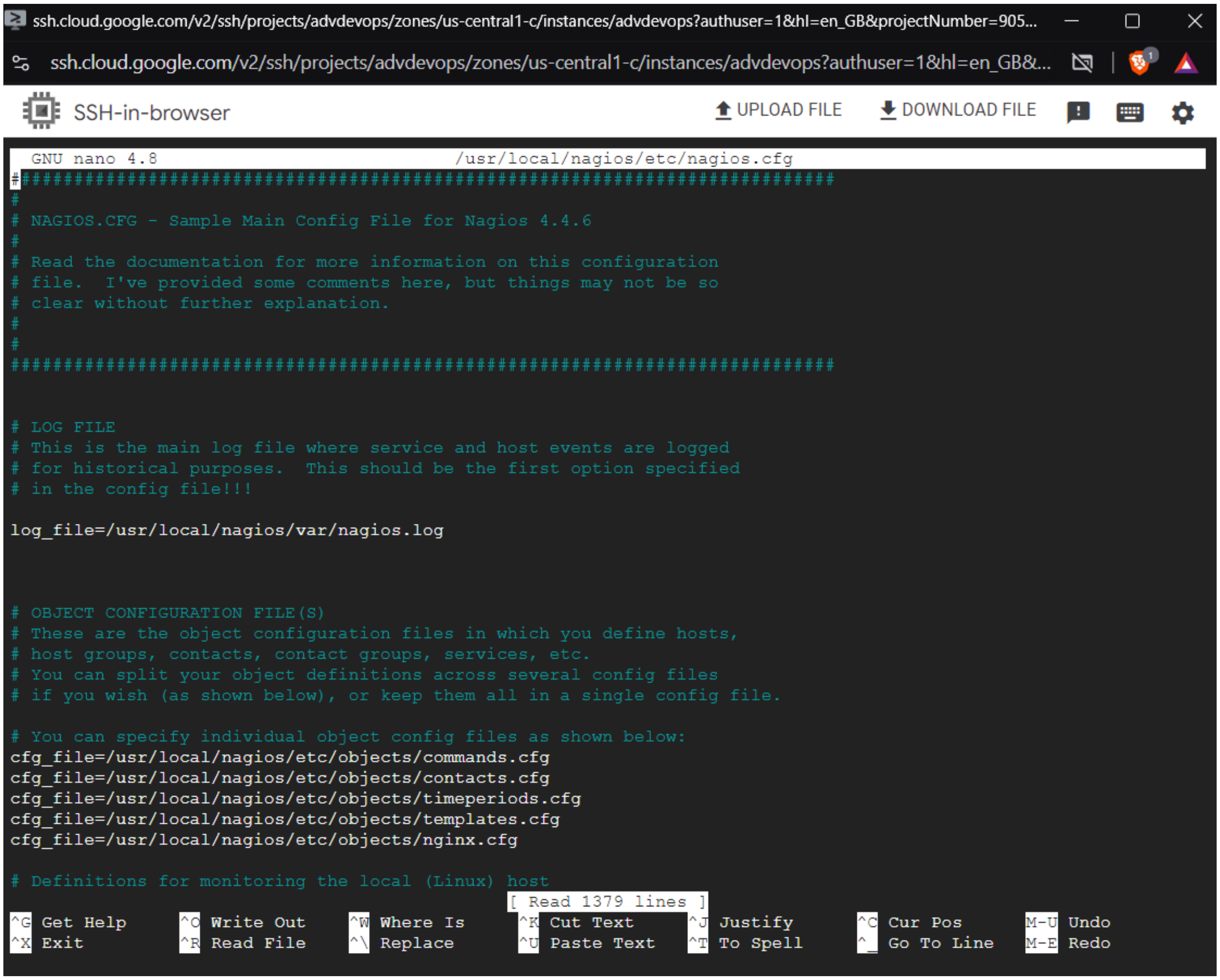
```
sudo mkdir -p /usr/local/nagios/etc/servers
sudo nano /usr/local/nagios/etc/servers/nginx.cfg
```



ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&projectNumber=905...
ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&...
SSH-in-browser
UPLOAD FILE
DOWNLOAD FILE

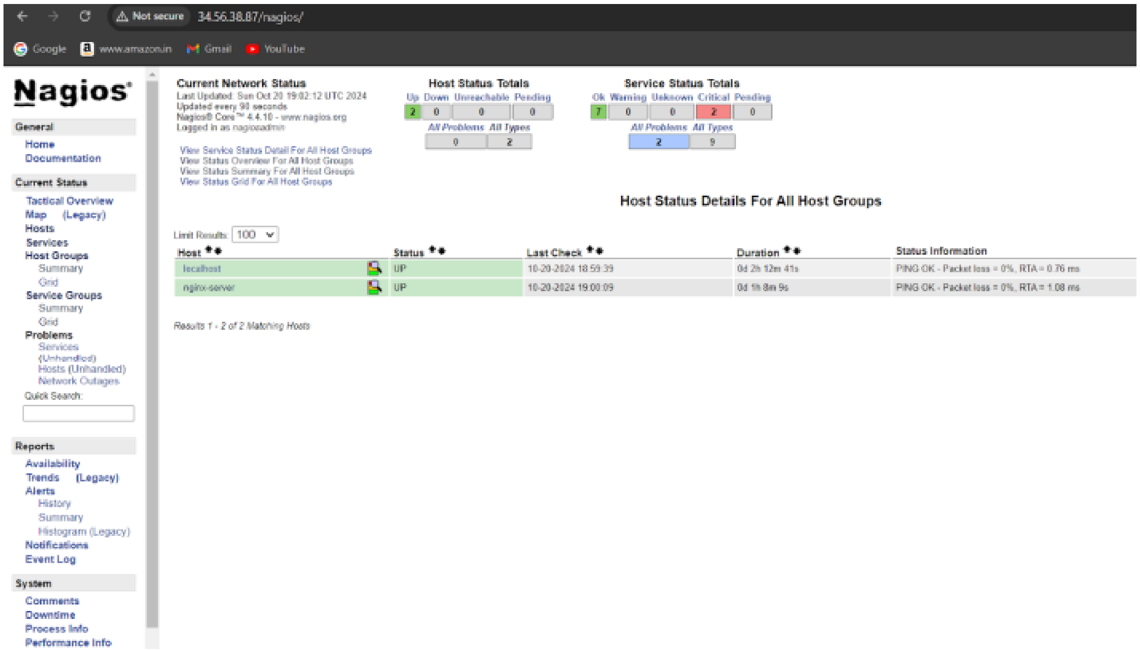
```
  GNU nano 4.8                    /usr/local/nagios/etc/servers/nginx.cfg                    Modified
define host {
use linux-server
host_name nginx-server
address <EXTERNAL_IP_OF_NGINX>//your ip addressof nignix
max_check_attempts 5
check_period 24x7
notification_interval 30
notification_period 24x7
}
define service {
use generic-service
host_name nginx-server
service_description HTTP
check_command check_http
notifications_enabled 1
}
```

Step 20 Add this line
cfg_ ile=/usr/local/nagios/etc/objects/nginx.cfg
In
sudo nano /usr/local/nagios/etc/nagios.cfg


ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&projectNumber=905...
ssh.cloud.google.com/v2/ssh/projects/advdevops/zones/us-central1-c/instances/advdevops?authuser=1&hl=en_GB&...
SSH-in-browser
UPLOAD FILE
DOWNLOAD FILE

```
  GNU nano 4.8                          /usr/local/nagios/etc/nagios.cfg
##############################################################################
#
# NAGIOS.CFG - Sample Main Config File for Nagios 4.4.6
#
# Read the documentation for more information on this configuration
# file.  I've provided some comments here, but things may not be so
# clear without further explanation.
#
#
##############################################################################


# LOG FILE
# This is the main log file where service and host events are logged
# for historical purposes.  This should be the first option specified
# in the config file!!!

log_file=/usr/local/nagios/var/nagios.log



# OBJECT CONFIGURATION FILE(S)
# These are the object configuration files in which you define hosts,
# host groups, contacts, contact groups, services, etc.
# You can split your object definitions across several config files
# if you wish (as shown below), or keep them all in a single config file.

# You can specify individual object config files as shown below:
cfg_file=/usr/local/nagios/etc/objects/commands.cfg
cfg_file=/usr/local/nagios/etc/objects/contacts.cfg
cfg_file=/usr/local/nagios/etc/objects/timeperiods.cfg
cfg_file=/usr/local/nagios/etc/objects/templates.cfg
cfg_file=/usr/local/nagios/etc/objects/nginx.cfg

# Definitions for monitoring the local (Linux) host
                                [ Read 1379 lines ]
^G Get Help     ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos      M-U Undo
^X Exit         ^R Read File    ^\ Replace      ^U Paste Text   ^T To Spell     ^_ Go To Line   M-E Redo
```

Step 21:Now you will be able to monitor pods:



Current Network Status
Last Updated: Sun Oct 20 19:02:12 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.4.10 - www.nagios.org
Logged in as nagiosadmin

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
7	0	0	2	0

All Problems	All Types
2	9

Host Status Details For All Host Groups

Limit Results: 100

Host	Status	Last Check	Duration	Status Information
localhost	UP	10-20-2024 18:59:39	0d 2h 12m 41s	PING OK - Packet loss = 0%, RTA = 0.76 ms
nginx-server	UP	10-20-2024 19:00:09	0d 1h 8m 9s	PING OK - Packet loss = 0%, RTA = 1.08 ms

Results 1 - 2 of 2 Matching Hosts

5.Conclusion:

The deployment of a basic Nginx server on a Kubernetes cluster, utilizing Google Cloud Console, was carried out successfully. This replaced the originally intended AWS Cloud9 environment, which was no longer supported for Kubernetes-based deployments. The process involved setting up a Kubernetes cluster, deploying the Nginx application, and configuring Nagios for monitoring the application's health.

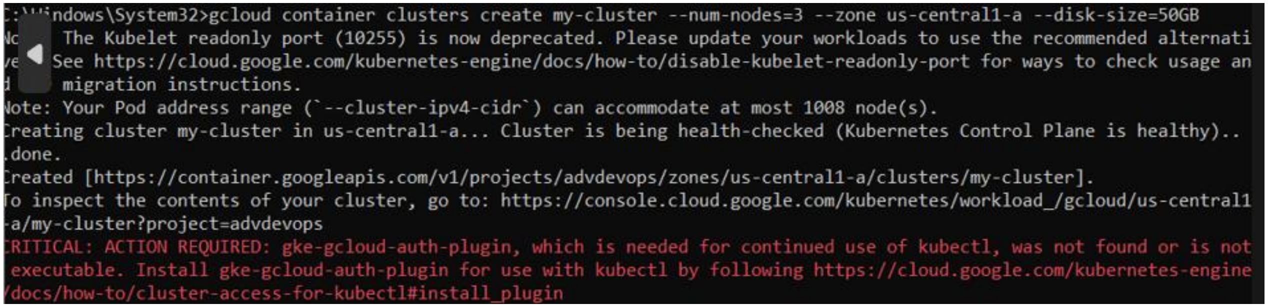
1. **Nginx Deployment:**
   An Nginx server was successfully deployed within the Kubernetes cluster, showcasing the platform's ability to efficiently manage containerized applications.

2. **Nagios Configuration:**
   Nagios was installed and configured to monitor the Nginx pod's availability.

3. **Health Monitoring Verification:**
   The monitoring system was thoroughly tested, with Nagios successfully detecting the Nginx pod's state.

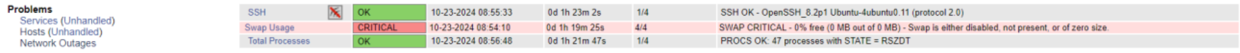
Problems I Faced:

1. The google cloud plugin which I downloaded earlier was a depreciated one so I had to re download the plugin and authenticate it with my account



```
:\\indows\System32>gcloud container clusters create my-cluster --num-nodes=3 --zone us-central1-a --disk-size=50GB
Nc   The Kubelet readonly port (10255) is now deprecated. Please update your workloads to use the recommended alternati
ve  See https://cloud.google.com/kubernetes-engine/docs/how-to/disable-kubelet-readonly-port for ways to check usage an
d    migration instructions.
Note: Your Pod address range (`--cluster-ipv4-cidr`) can accommodate at most 1008 node(s).
Creating cluster my-cluster in us-central1-a... Cluster is being health-checked (Kubernetes Control Plane is healthy)..
.done.
Created [https://container.googleapis.com/v1/projects/advdevops/zones/us-central1-a/clusters/my-cluster].
To inspect the contents of your cluster, go to: https://console.cloud.google.com/kubernetes/workload_/gcloud/us-central1
-a/my-cluster?project=advdevops
CRITICAL: ACTION REQUIRED: gke-gcloud-auth-plugin, which is needed for continued use of kubectl, was not found or is not
 executable. Install gke-gcloud-auth-plugin for use with kubectl by following https://cloud.google.com/kubernetes-engine
/docs/how-to/cluster-access-for-kubectl#install_plugin
```

2. Another issue was the Swap Memory Error which was there there was no space left, which was disabled and was reading it as “Not Available”



Problems
Services (Unhandled)
Hosts (Unhandled)
Network Outages

SSH	☒	OK	10-23-2024 08:55:33	0d 1h 23m 2s	1/4	SSH OK - OpenSSH_8.2p1 Ubuntu-4ubuntu0.11 (protocol 2.0)
Swap Usage		CRITICAL	10-23-2024 08:54:10	0d 1h 19m 25s	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	10-23-2024 08:56:48	0d 1h 21m 47s	1/4	PROCS OK: 47 processes with STATE = RSZDT

3. While writing and creating the Nagios configuration file for Ngnix we must make sure the external IP address is mentioned correctly and file address is correctly mentioned in the main configuration file Of Nagios
4. Last error was in the Apache Server as I was opening the Nagios UI when I was clink on any element instead of displaying any information it was downloading a .cgi file, then I enabled the cgi modul in apache by “sudo a2enmod cgi” and restarted the apache “sudo systemctl restart apache2”