






Google Cloud Platform Project

Creating Apache Web Server on Google Cloud

Creator Information	
	Sultan Alanazi
	https://www.linkedin.com/in/sultan-la-alanazi/
	https://github.com/SultanAlanazie

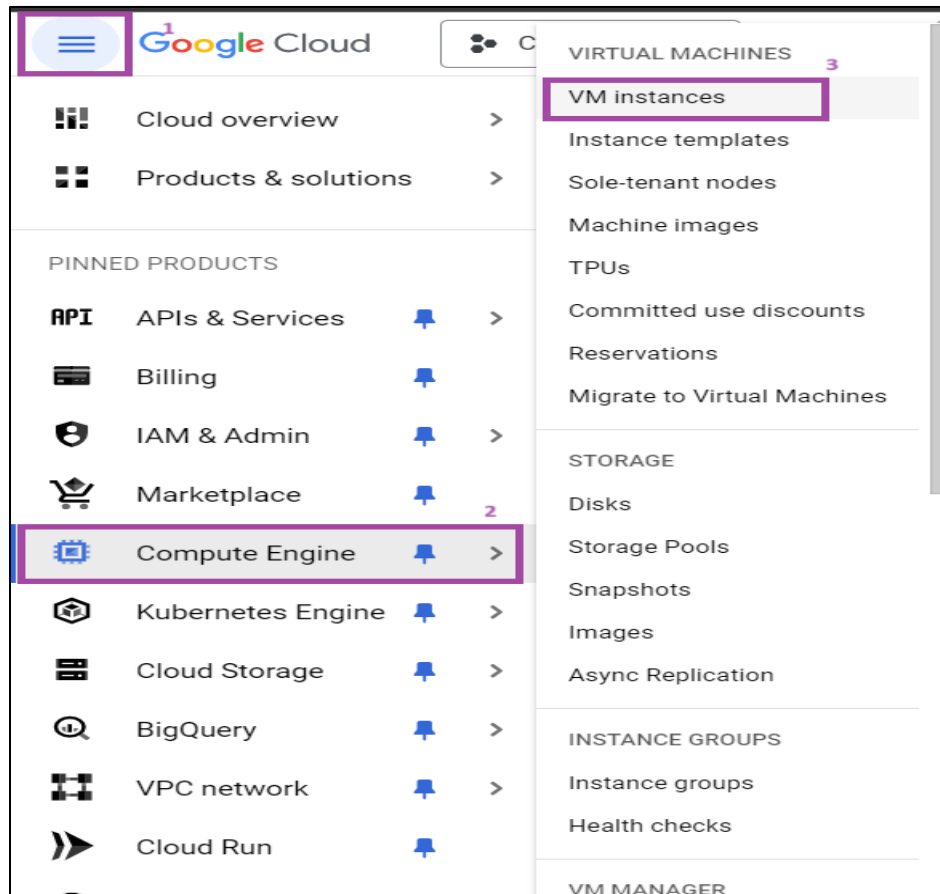
Contents

General Instructions.....	3
Setting up the VM Instance	4
Create a Virtual Machine (VM).....	4
Setting up VM SSH	5
Running the server.....	7

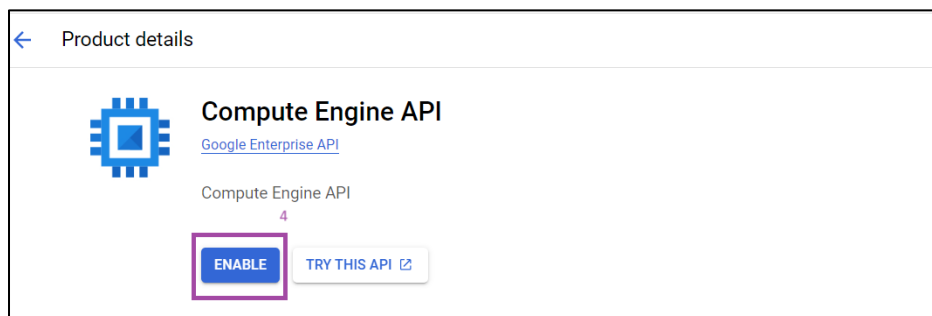


General Instructions

- 1- Make your own GCP account.
Through URL: <https://cloud.google.com/>
- 2- From The Navigation Menu, go to Compute Engine.
- 3- Create a virtual machine by choosing VM Instance.



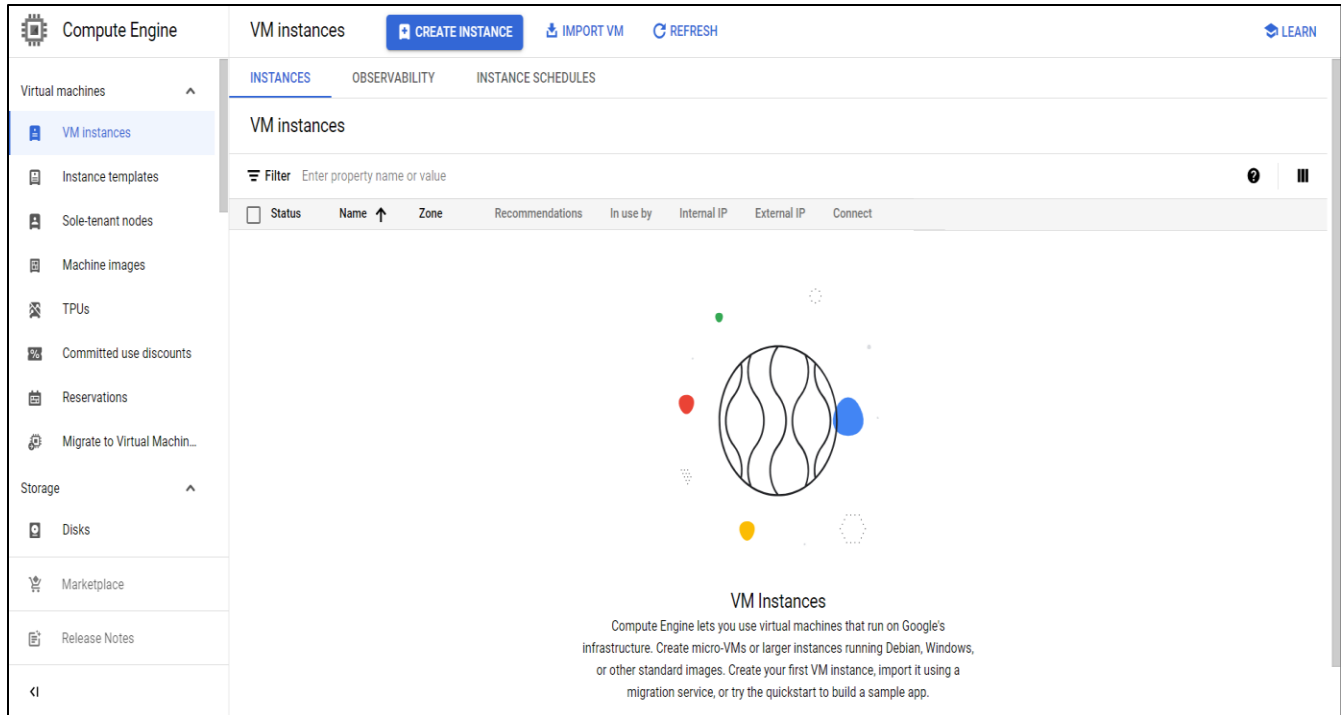
- 4- Enable Compute Engine



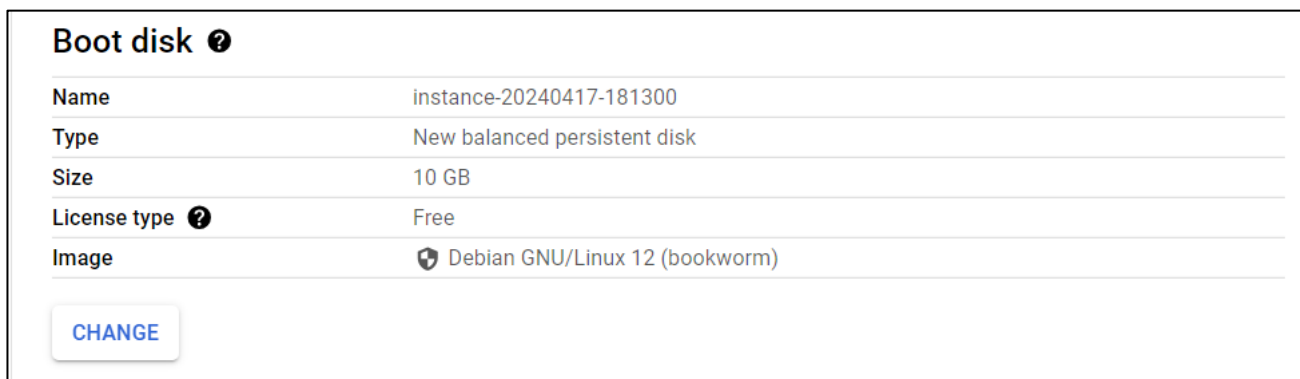
Setting up the VM Instance

Create a Virtual Machine (VM)

1. Press **“CREATE INSTANCE”**.



2. View The boot disk default settings (Can be changed by pressing the **“Change”** button).



3. FireWall section: Allow HTTP & HTTPS traffic.

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

Setting up VM SSH

1. run SSH by clicking the “SSH” button.

VM instances

CREATE INSTANCEIMPORT VMREFRESH

INSTANCESOBSERVABILITYINSTANCE SCHEDULES

VM instances

Filter Enter property name or value

<input type="checkbox"/> Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input checked="" type="checkbox"/>	web-server						SSH

2. An SSH window will open.

3. Update packages using the Command Line by writing:

VM instances

CREATE INSTANCE

INSTANCESOBSERVABILITYINSTANCE SCHEDULES

VM instances

Filter Enter property name or value

<input type="checkbox"/> Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input checked="" type="checkbox"/>	web-server						SSH

Related actions

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

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

SSH-in-browser

Linux web-server.us-centrall1-a.c.argon-triode-420514.internal 6.1.0-20-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.85-1 (2024-04-11) x86_64

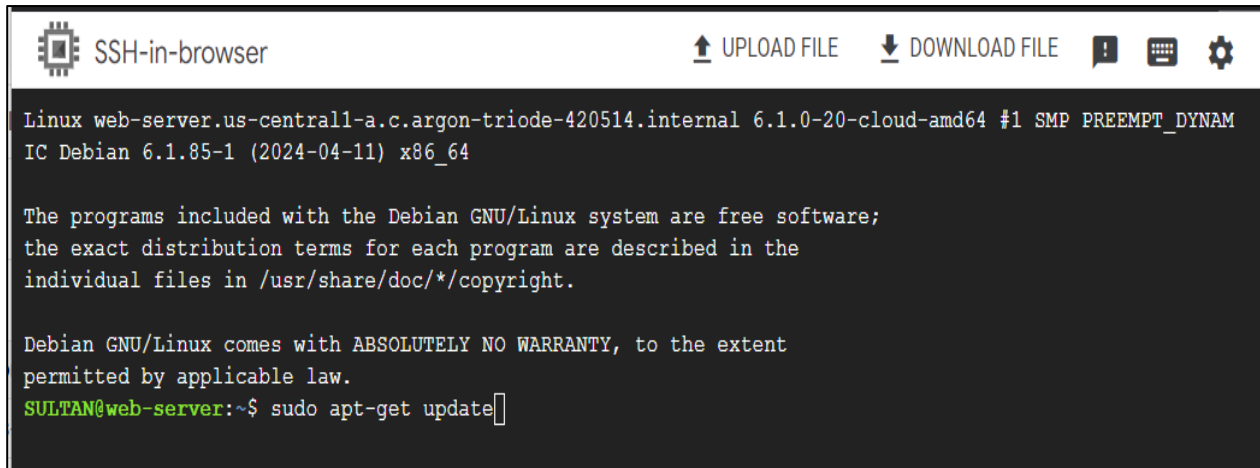
The programs included with the Debian GNU/Linux system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

SULTAN@web-server:~\$



sudo apt-get update



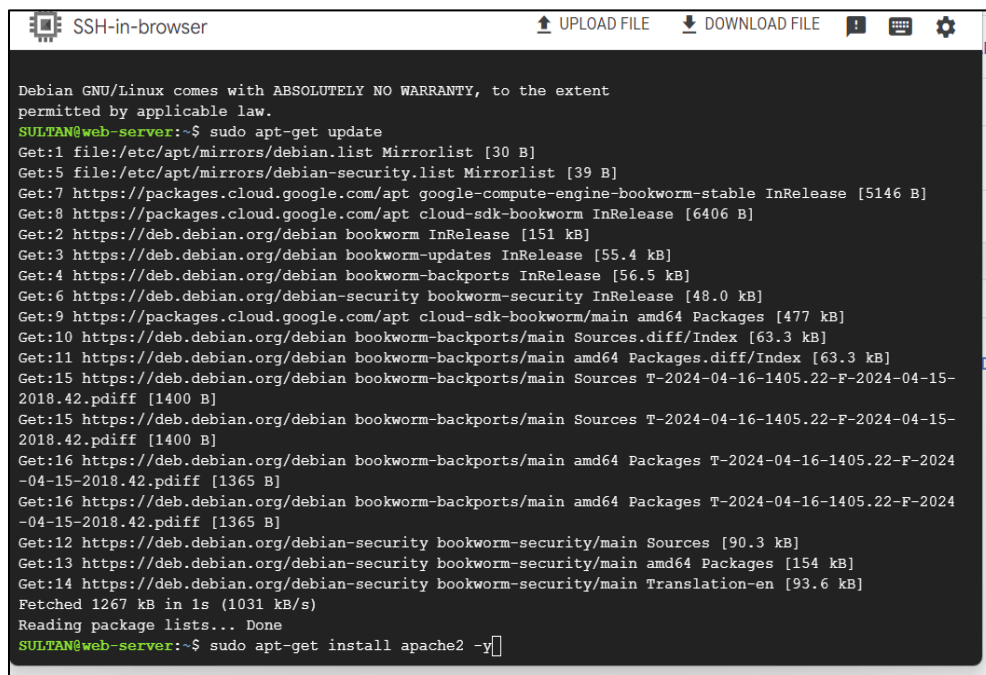
```
Linux web-server.us-central1-a.c.argon-triode-420514.internal 6.1.0-20-cloud-amd64 #1 SMP PREEMPT_DYNAM
IC Debian 6.1.85-1 (2024-04-11) x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
SULTAN@web-server:~$ sudo apt-get update
```

4. Install apache2 server using the Command Line by writing:

sudo apt-get install apache2 -y



```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
SULTAN@web-server:~$ sudo apt-get update
Get:1 file:/etc/apt/mirrors/debian.list Mirrorlist [30 B]
Get:5 file:/etc/apt/mirrors/debian-security.list Mirrorlist [39 B]
Get:7 https://packages.cloud.google.com/apt google-compute-engine-bookworm-stable InRelease [5146 B]
Get:8 https://packages.cloud.google.com/apt cloud-sdk-bookworm InRelease [6406 B]
Get:2 https://deb.debian.org/debian bookworm InRelease [151 kB]
Get:3 https://deb.debian.org/debian bookworm-updates InRelease [55.4 kB]
Get:4 https://deb.debian.org/debian bookworm-backports InRelease [56.5 kB]
Get:6 https://deb.debian.org/debian-security bookworm-security InRelease [48.0 kB]
Get:9 https://packages.cloud.google.com/apt cloud-sdk-bookworm/main amd64 Packages [477 kB]
Get:10 https://deb.debian.org/debian bookworm-backports/main Sources.diff/Index [63.3 kB]
Get:11 https://deb.debian.org/debian bookworm-backports/main amd64 Packages.diff/Index [63.3 kB]
Get:15 https://deb.debian.org/debian bookworm-backports/main Sources T-2024-04-16-1405.22-F-2024-04-15-
2018.42.pdiff [1400 B]
Get:15 https://deb.debian.org/debian bookworm-backports/main Sources T-2024-04-16-1405.22-F-2024-04-15-
2018.42.pdiff [1400 B]
Get:16 https://deb.debian.org/debian bookworm-backports/main amd64 Packages T-2024-04-16-1405.22-F-2024
-04-15-2018.42.pdiff [1365 B]
Get:16 https://deb.debian.org/debian bookworm-backports/main amd64 Packages T-2024-04-16-1405.22-F-2024
-04-15-2018.42.pdiff [1365 B]
Get:12 https://deb.debian.org/debian-security bookworm-security/main Sources [90.3 kB]
Get:13 https://deb.debian.org/debian-security bookworm-security/main amd64 Packages [154 kB]
Get:14 https://deb.debian.org/debian-security bookworm-security/main Translation-en [93.6 kB]
Fetched 1267 kB in 1s (1031 kB/s)
Reading package lists... Done
SULTAN@web-server:~$ sudo apt-get install apache2 -y
```

Now the Server is installed on the VM.



Running the server

1. To check if the server is working use the Command Line by writing :
`sudo service --status-all`

```
SSH-in-browser  UPLOAD FILE  DOWNLOAD FILE  !  [ ]  [ ]  [ ]
Enabling module authn_core.
Enabling module auth_basic.
Enabling module access_compat.
Enabling module authn_file.
Enabling module authz_user.
Enabling module alias.
Enabling module dir.
Enabling module autoindex.
Enabling module env.
Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /lib/systemd/system/apache-htcacheclean.service.
Processing triggers for man-db (2.11.2-2) ...
Processing triggers for libc-bin (2.36-9+deb12u4) ...
SULTAN@web-server:~$ sudo service --status-all
```

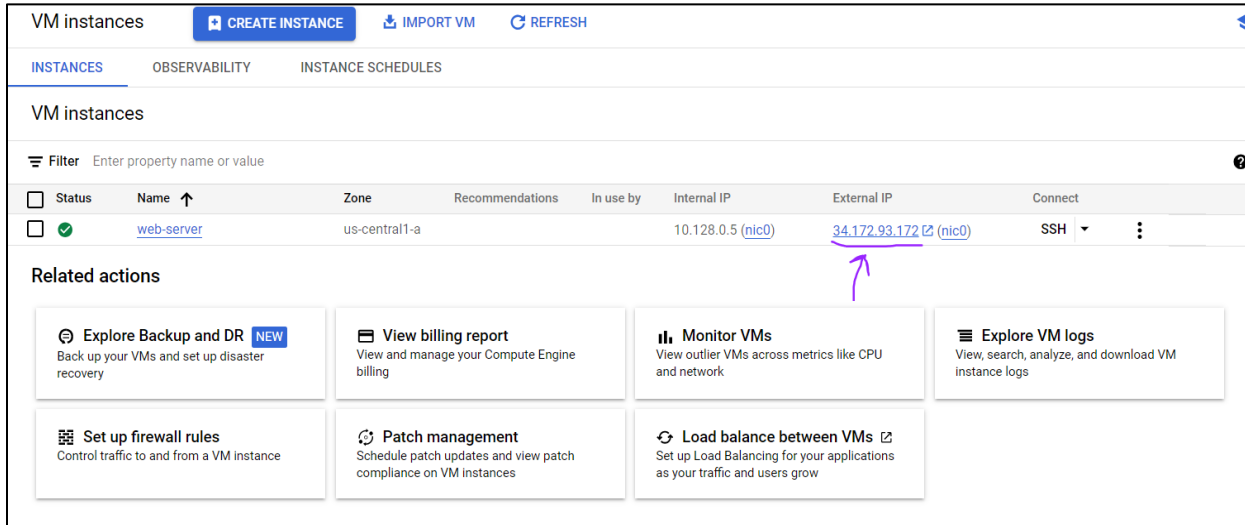
```
SSH-in-browser  UPLOAD FILE  DOWNLOAD FILE  !  [ ]  [ ]  [ ]
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /lib/systemd/system/apache-htcacheclean.service.
Processing triggers for man-db (2.11.2-2) ...
Processing triggers for libc-bin (2.36-9+deb12u4) ...
SULTAN@web-server:~$ sudo service --status-all
[ + ] apache-htcacheclean
[ + ] apache2
[ + ] apparmor
[ + ] cron
[ + ] dbus
[ + ] exim4
[ + ] haveged
[ - ] hwclock.sh
[ + ] kmod
[ + ] procps
[ - ] screen-cleanup
[ + ] ssh
[ - ] sudo
[ + ] udev
[ + ] unattended-upgrades
[ - ] uuidd
SULTAN@web-server:~$
```



Note:

Apache By default will provide a home page named “**index.html**”

You can access it by clicking on the “**External IP**” button at the VMInstances page.



VM instances

CREATE INSTANCE IMPORT VM REFRESH

INSTANCES OBSERVABILITY INSTANCE SCHEDULES

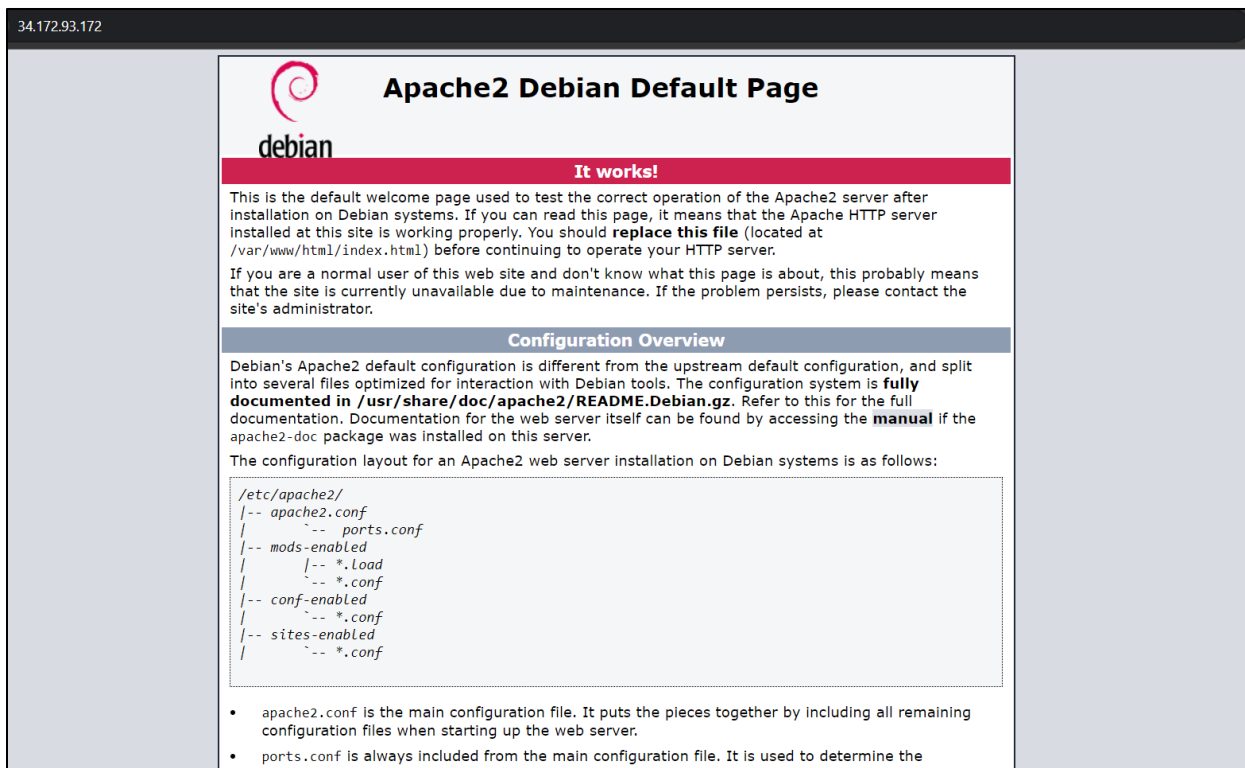
VM instances

Filter Enter property name or value

Status	Name	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input checked="" type="checkbox"/>	web-server	us-central1-a			10.128.0.5 (nic0)	34.172.93.172 (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, analyze, 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



34.172.93.172

Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.



2. To use a custom Home Page, make sure to name the file “index.html” and locate it in the following path :

/var/www/html

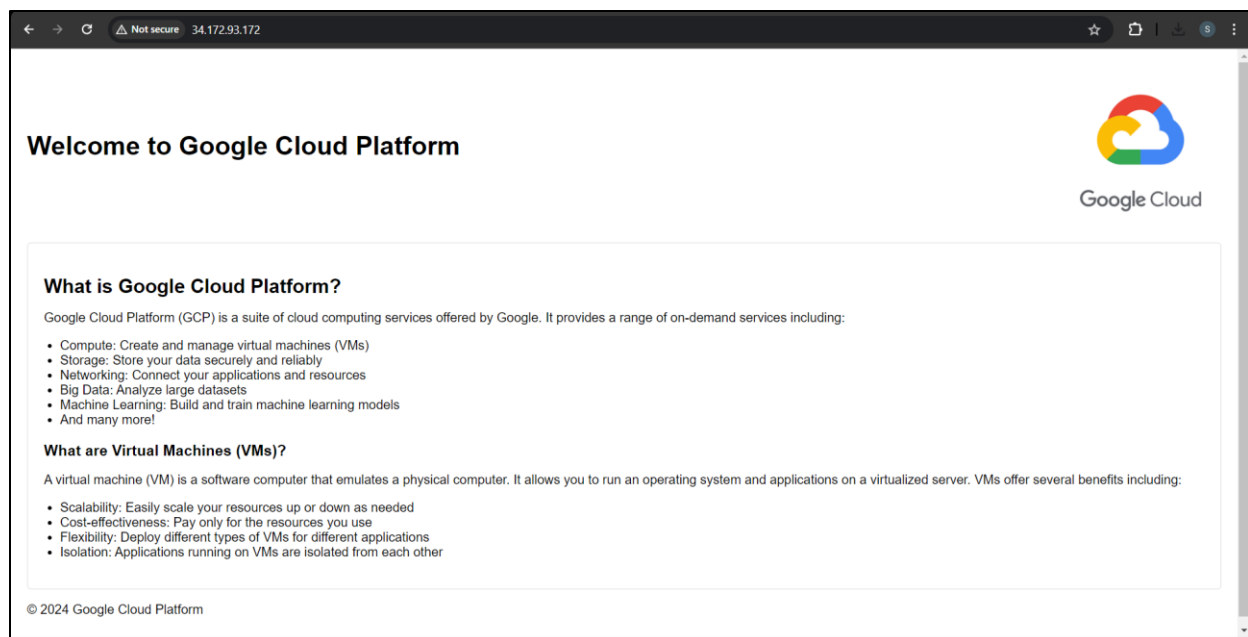
example:

my index.html file is presented at the root folder for the web server.

The following screenshot presents the “move” command to locate the file at the desired destination.

```
SULTAN@web-server:~$ sudo mv index.html /var/www/html
SULTAN@web-server:~$ ls
SULTAN@web-server:~$ cd /var/www/html
SULTAN@web-server:/var/www/html$ ls
index.html
SULTAN@web-server:/var/www/html$
```

The following screenshot shows the custom “index.html” file we moved to the /html path.



Good Luck!



Created With Passion
By **SULTAN**

