How to Make a Linux User Have the sudo Function?

written by sysadmin | 5 February 2025

SUDO stands for "**SuperUser DO**" and it is a program for Unix-like computer operating systems that enables users to run programs with the security privileges of another user, by default, the superuser. With sudo, a normal user can install or delete an application, change the server network, or even reboot or shut down the server.

Problem

How to make a Linux user have the sudo function?

Solution

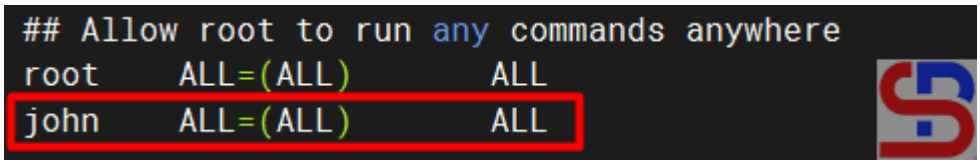
This article will explain how to make a Linux user have the sudo function on RockyLinux/AlmaLinux/CentOS, Ubuntu/Debian, and OpenSUSE distros. For example, you want to add the user john to these distros and want that user to be able to use the sudo function. As far as I know, there are two methods to do it:

1. Change the sudoers file

Open the /etc/sudoers file or use the command below:

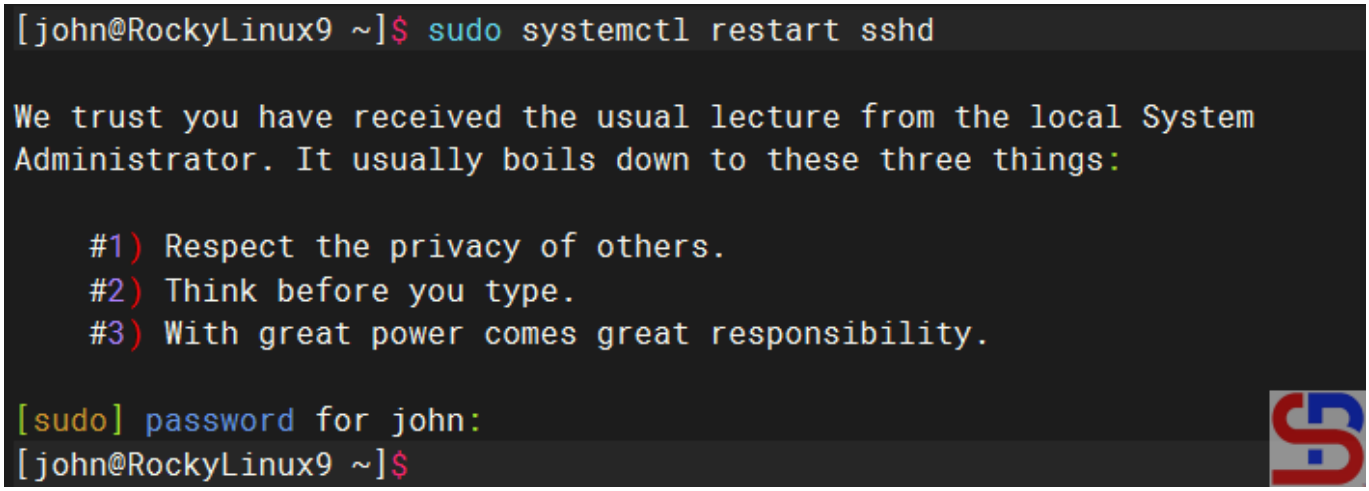
```
visudo
```

Add to the file the user name as in the image below:



Add the user in the sudoers file

After that, save the file and then try to add a new user using the user john, if there is a display like the image below:

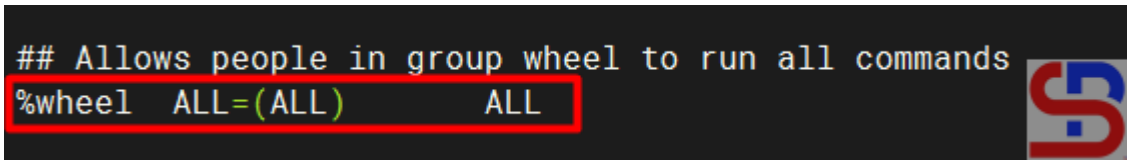


Choose number 1

Then select number **1**, and the user should successfully add a new user as in the image above.

2. Add the user to the sudo group

Add the user to the sudo group, where the name of this sudo group can vary in each distro. To see the name of the sudo group, look in the sudoers file and look for a sentence similar to '**Allows people in group to execute any command**'. For example, in RockyLinux and OpenSUSE, the name of the sudo group is **wheel**, **sudo** in Ubuntu, and don't forget to make sure to uncomment the section as in the image below:



Check the sudo group in the sudoers file

Then type the command below so that a user can use sudo:

RockyLinux & OpenSUSE

```
usermod -aG wheel john
```

```
[root@RockyLinux9 ~]# usermod -aG wheel john
[root@RockyLinux9 ~]#
[root@RockyLinux9 ~]# su - john
Last login: Wed Jan 15 05:51:59 EST 2025 on pts/0
[john@RockyLinux9 ~]$ sudo adduser edward
[sudo] password for john:
[john@RockyLinux9 ~]$
```

Add the user to the sudo group

Ubuntu/Debian

```
usermod -aG sudo john
```

Note

The two methods above can provide the sudo feature to a user on Linux so that the user can run commands that can only be executed by root if the user uses the sudo command by writing down the password. However, if you want the bob user not to have to enter a password when running the sudo command, then in the sudoers file, type the script below:

```
bob             ALL=(ALL)       NOPASSWD: ALL
```

Use the command below if you want the robin user to only be able to perform reboot commands using sudo, but not other commands using sudo:

```
robin           ALL=(ALL)     /usr/sbin/reboot
```

```
[robin@RockyLinux9 ~]$ sudo systemctl restart sshd
[sudo] password for robin:
Sorry, user robin is not allowed to execute '/bin/systemctl restart sshd' as root on RockyLinux9.
[robin@RockyLinux9 ~]$
```

Give the partial sudo function to the user

References

en.wikipedia.org
askubuntu.com
phoenixnap.com

How to Display the Timestamp in the History Command?

written by sysadmin | 5 February 2025

Displaying the timestamp in the history command is very useful for various purposes. However, in general, Linux systems do not display a timestamp when you run the history command

Problem

How to display the timestamp in the history command?

Solution

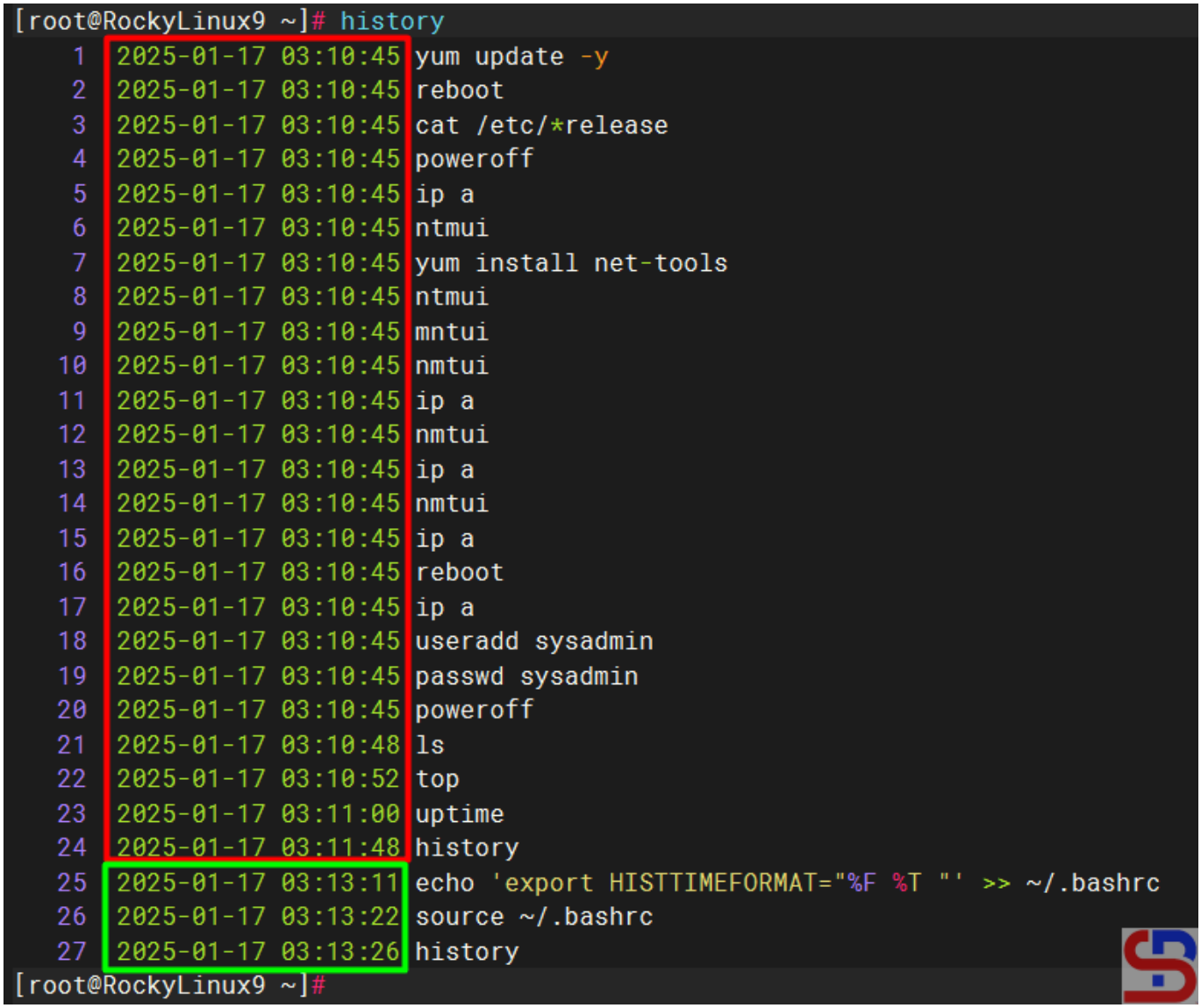
If you type the history command on your Linux server, by default, you will find that there is no timestamp, as in the image below:

```
[root@RockyLinux9 ~]# history
    1  yum update -y
    2  reboot
    3  cat /etc/*release
    4  poweroff
    5  ip a
    6  ntmui
    7  yum install net-tools
    8  ntmui
    9  mntui
   10  nmtui
   11  ip a
   12  nmtui
   13  ip a
   14  nmtui
   15  ip a
   16  reboot
   17  ip a
   18  useradd sysadmin
   19  passwd sysadmin
   20  poweroff
   21  ls
   22  top
   23  uptime
   24  history
[root@RockyLinux9 ~]#
```

The history command

So that your Linux server can display timestamps in the history command, type the command below:

```
echo 'export HISTTIMEFORMAT="%F %T "' >> ~/.bashrc
source ~/.bashrc
```



The history command with a timestamp

The image above shows that the timestamp is already visible when you type the history command. Linux commands executed for a long time will display the same timestamp (look at the image above in the red box). However, if you run another Linux command, the timestamp displayed will be the same as when you executed the Linux command (look at the image above in the green box).

Note

By default, you have to run the commands above on each user to display the timestamps in the history command. But I think it’s very tiring to do that. So, if you want to

display a timestamp in the history command for each Linux user, copy the command below:

```
sudo vi /etc/profile.d/history-timestamp.sh
```

After that, copy the script below into the file:

```
export HISTTIMEFORMAT="%F %T "
```

and then run the below script:

```
sudo chmod 644 /etc/profile.d/history-timestamp.sh
```

The history command in the new user's shell will display timestamps automatically if there is a new user on your Linux server.

References

cyberciti.biz
stackoverflow.com
tecmint.com
linuxhandbook.com

How to Install gcloud on a Linux Server?

written by sysadmin | 5 February 2025

The previous articles explained how to install gcloud on Ubuntu/Debian distros and RockyLinux/AlmaLinux/CentOS distros. This article will explain how to install gcloud on Linux.

Problem

How to install gcloud on a Linux server?

Solution

If you use Linux other than the Ubuntu/Debian distro and the RockyLinux/AlmaLinux/CentOS distro, and you want to install gcloud on your Linux distro, then below are the steps (I use OpenSUSE 15 distro):

A. Install gcloud

As far as I know, there are 2 methods for installing on a Linux server, and both methods recommend using a user other than root.

1. Use the script

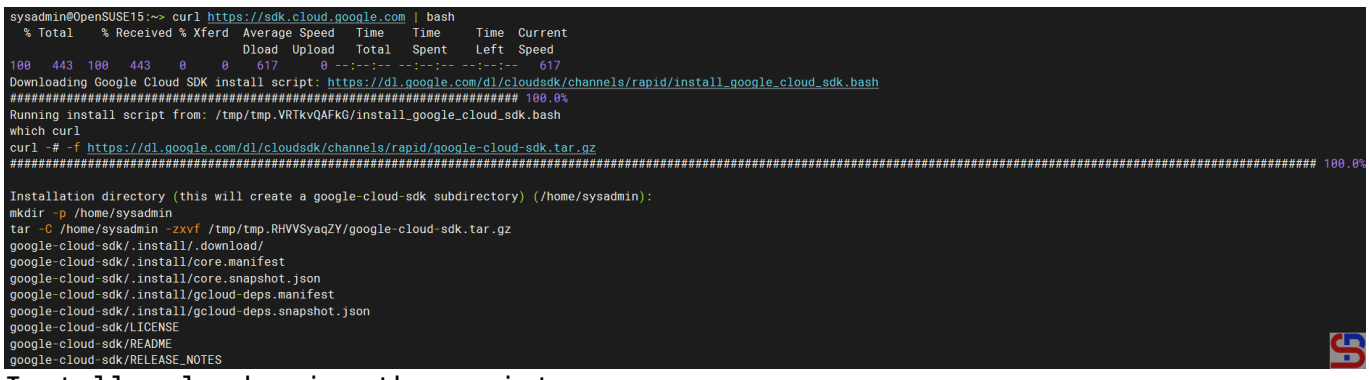
Before you install gcloud using the script, make sure there are tar and curl packages, and **Python version 3.8 and up** on your server. You can check it with the following command:

```
python3 --version
```

After that, use the following command to download and install the script:

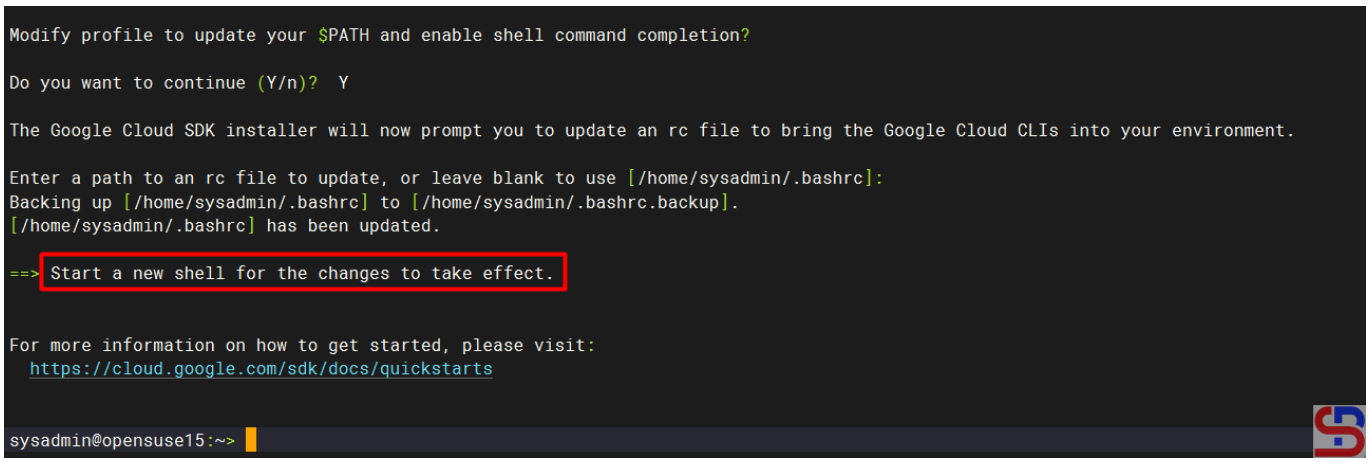
```
curl https://sdk.cloud.google.com | bash
```

Then you will see a display like the one below:



Install gcloud using the script

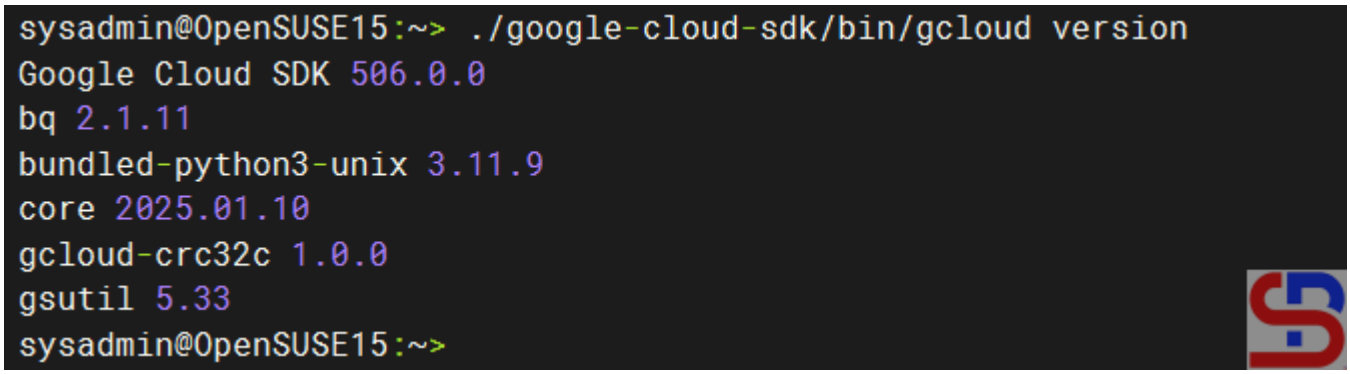
Wait until it’s finished, and you will see a display like the one below:



Installation complete

From the image above, you are asked to create a new SSH connection so that the effect can be seen, and type the command below:

```
./google-cloud-sdk/bin/gcloud version
```



Execute the gcloud version command

If you want to type the gcloud command without having to type **./google-cloud-sdk/bin/gcloud**, then run the command below:

```
echo "alias gcloud=./google-cloud-sdk/bin/gcloud" >> ~/.bashrc
source ~/.bashrc
```

```
sysadmin@OpenSUSE15:~> echo "alias gcloud=./google-cloud-sdk/bin/gcloud" >> ~/.bashrc
sysadmin@OpenSUSE15:~> source ~/.bashrc
sysadmin@OpenSUSE15:~>
sysadmin@OpenSUSE15:~> gcloud version
Google Cloud SDK 506.0.0
bq 2.1.11
bundled-python3-unix 3.11.9
core 2025.01.10
gcloud-crc32c 1.0.0
gsutil 5.33
sysadmin@OpenSUSE15:~>
```

Make an alias for gcloud

2. Using the installer

Run the following commands to install gcloud on your Linux server:

```
curl -O
https://dl.google.com/dl/cloudsdk/channels/rapid/downloads/google-cloud-cli-linux-x86_64.tar.gz
tar -xf google-cloud-cli-linux-x86_64.tar.gz
./google-cloud-sdk/install.sh
```

After installation completes, use the following command to test the gcloud command:

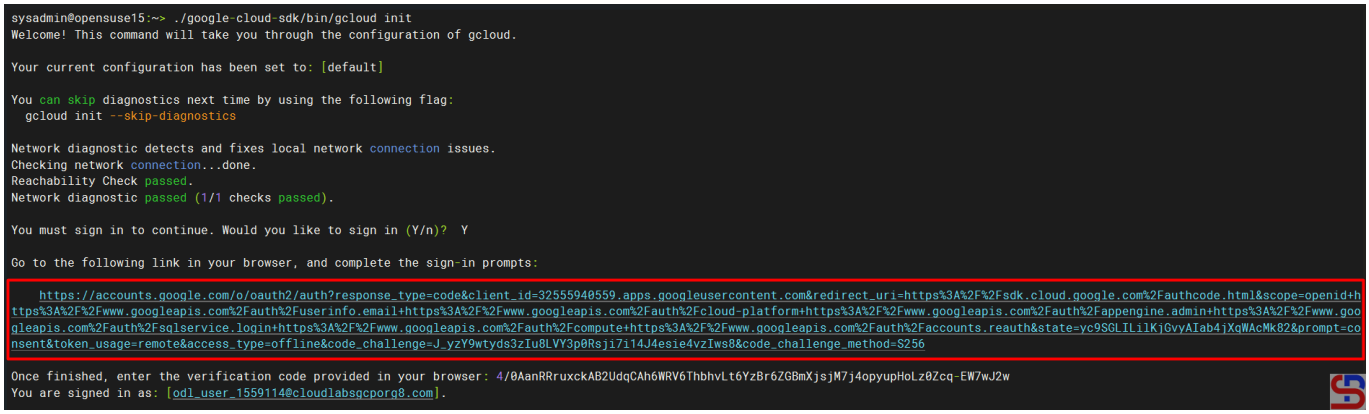
```
./google-cloud-sdk/bin/gcloud version
```

B. Connect to GCP

After you install gcloud on your server, type the command below:

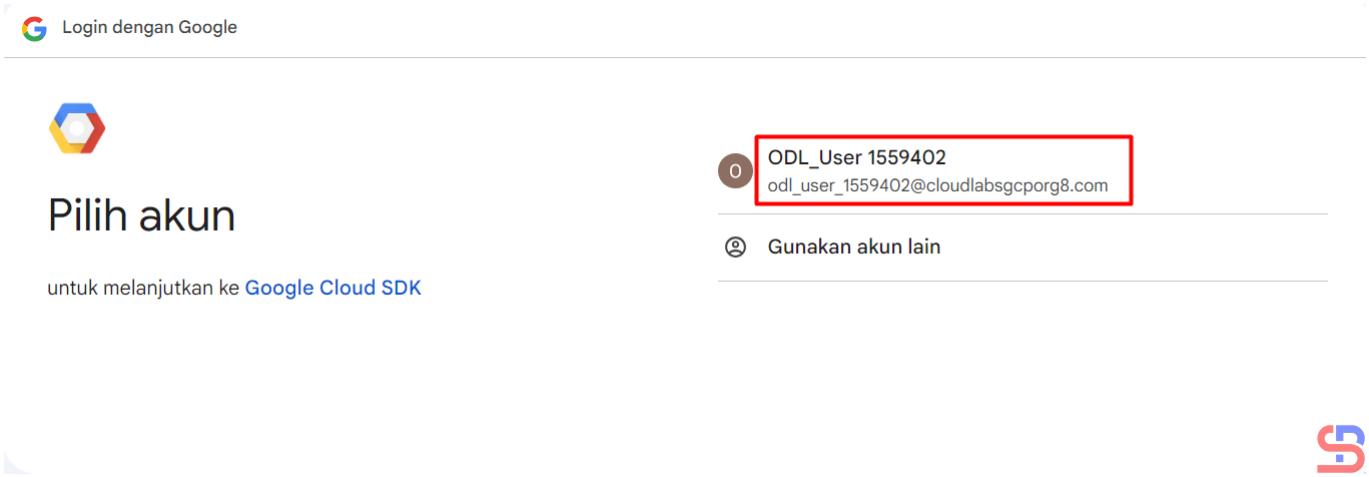
```
gcloud init
```

Then there will be a display like the image below:



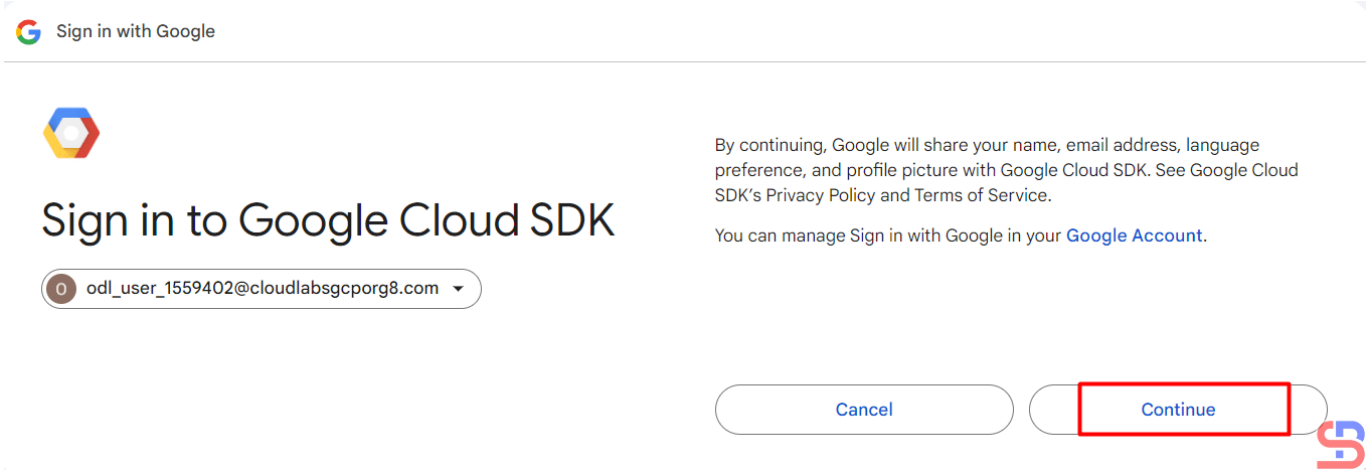
Click the link

Click the **Ctrl+Click** button in the red box to open the link in a browser, or if you have difficulty, copy what is in the red box and place it in your browser so you will see a display like the one below:



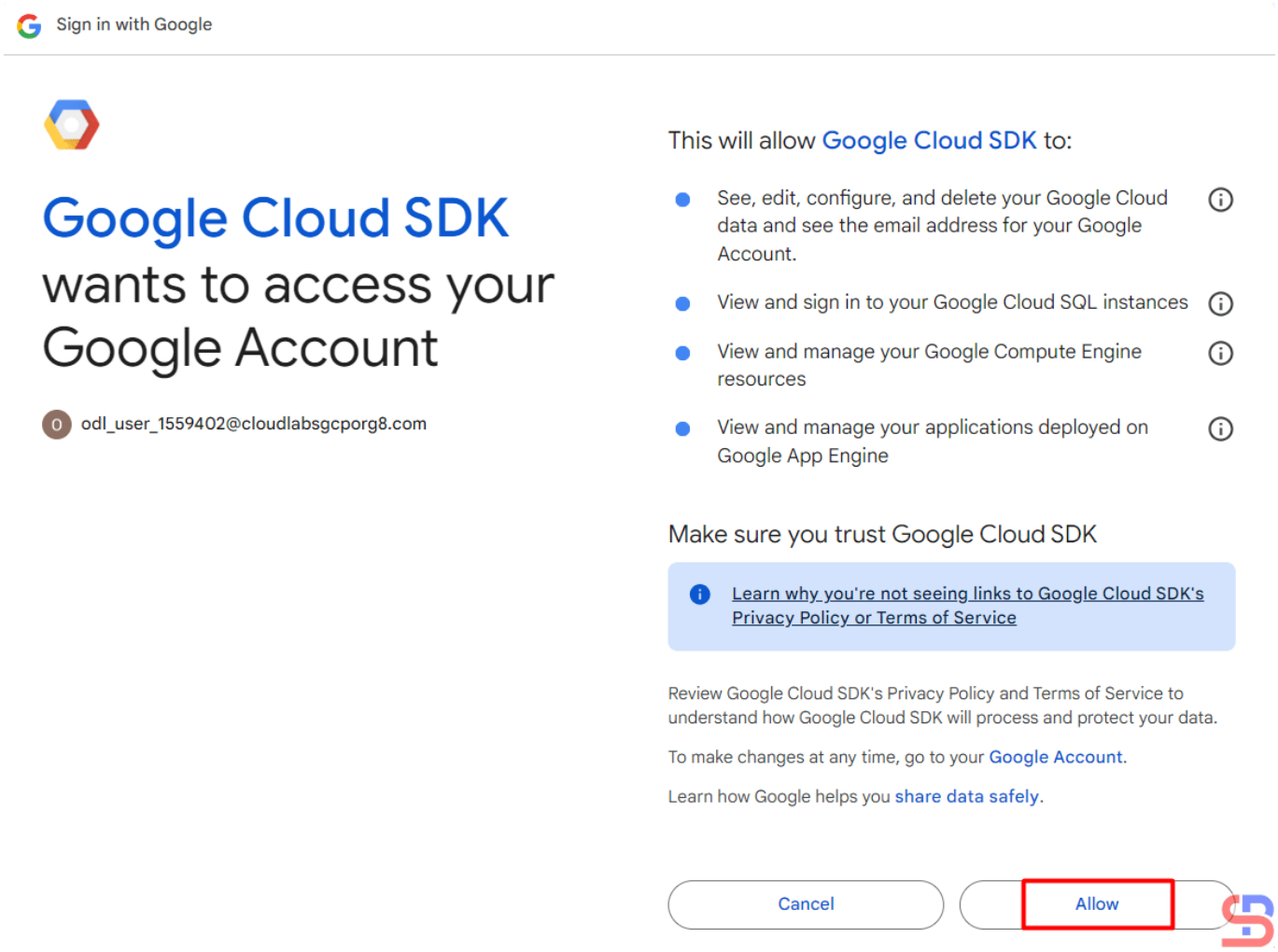
Click the account

Click on the Google account that will access GCP, then there will be a display like the image below:



Click the Continue button

Click the **Continue** button, then the display below will appear:



Click the Allow button

Click the **Allow** button, then the display below will appear:

Sign in to the gcloud CLI

You are seeing this page because you ran the following command in the gcloud CLI from this or another machine. If this is not the case, close this tab.

```
gcloud auth login --no-launch-browser
```

Enter the following verification code in gcloud CLI on the machine you want to log into. This is a credential **similar to your password** and should not be shared with others.

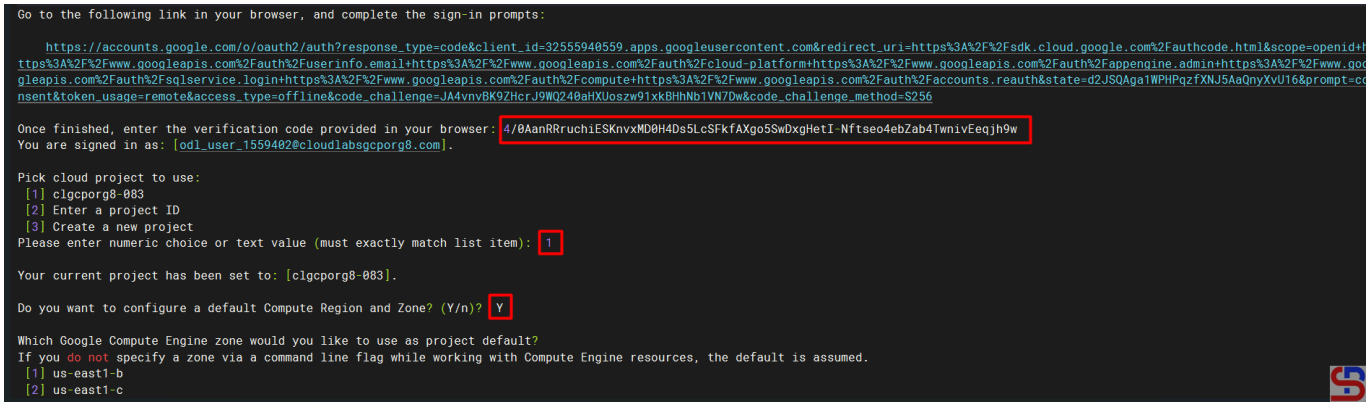
```
4/0AanRRruchiESKnvxMD0H4Ds5LcSFkfAXgo5
SwDxgHetI-Nftseo4ebZab4TwnivEeqjh9w
```

Copy

You can close this tab when you're done.

Click the Copy button

Click the **Copy** button, and paste it into the CLI on your server as in the image below:



Paste the code

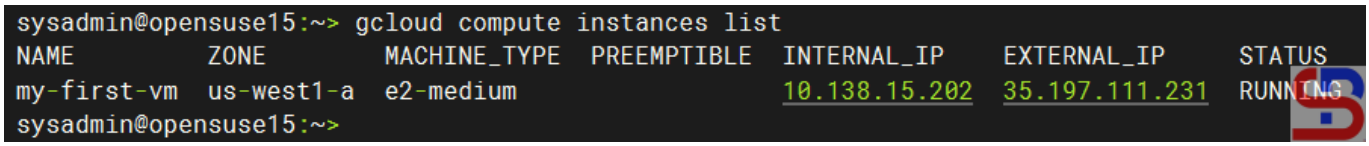
Select the project and configure the zone as in the image above. After that, the gcloud configuration is complete.

C. Test gcloud

Now, try gcloud to access your GCP. I try to list my virtual machine in GCP using the below command:

```
gcloud compute instances list
```

Then the display below will appear:



Display virtual machine in GCP using gcloud

If you get a display like the one above, you have successfully used gcloud to access your GCP.

Note

If you have many projects on your GCP, you can choose one of these projects as the starting point for your gcloud on GCP. You can switch projects using the command:

```
gcloud config set project PROJECT_ID
```

Change **PROJECT_ID** to the project ID you want to switch to.

References

cloud.google.com
liquidweb.com
bacancytechnology.com

How to Change SSH Port?

written by sysadmin | 5 February 2025

If you access a device such as a server using an SSH connection, you are using port 22 by default. However, port 22 is often the target of security attacks, so it is recommended that you change the SSH port.

Problem

How to change SSH Port?

Solution

To change the SSH port on a Linux server, go to the **/etc/ssh/sshd.config** file, look for the line containing Port 22 and set it to the number you want to change. For example, you want to change the SSH port to port 43210, so change the line as in the script below from:

```
#Port 22
```

to

```
Port 43210
```

After that, restart the SSH service using the command below:

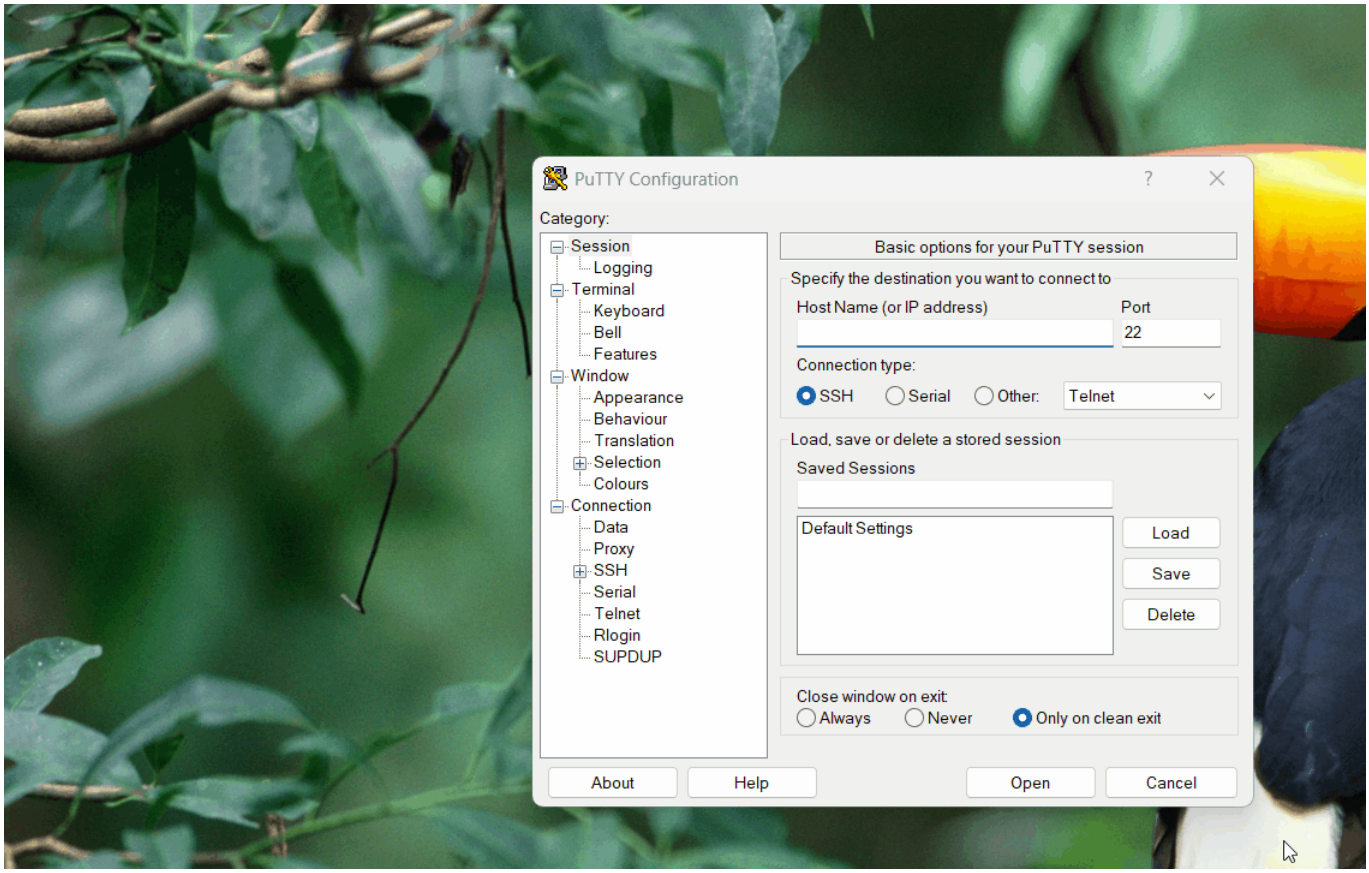
Ubuntu/Debian

```
systemctl restart ssh
```

RockyLinux/AlmaLinux/CentOS & OpenSUSE

```
systemctl restart sshd
```

After that, test by accessing SSH using port 43210. If you use Putty, then change port 22 to 43210 as in the image below:



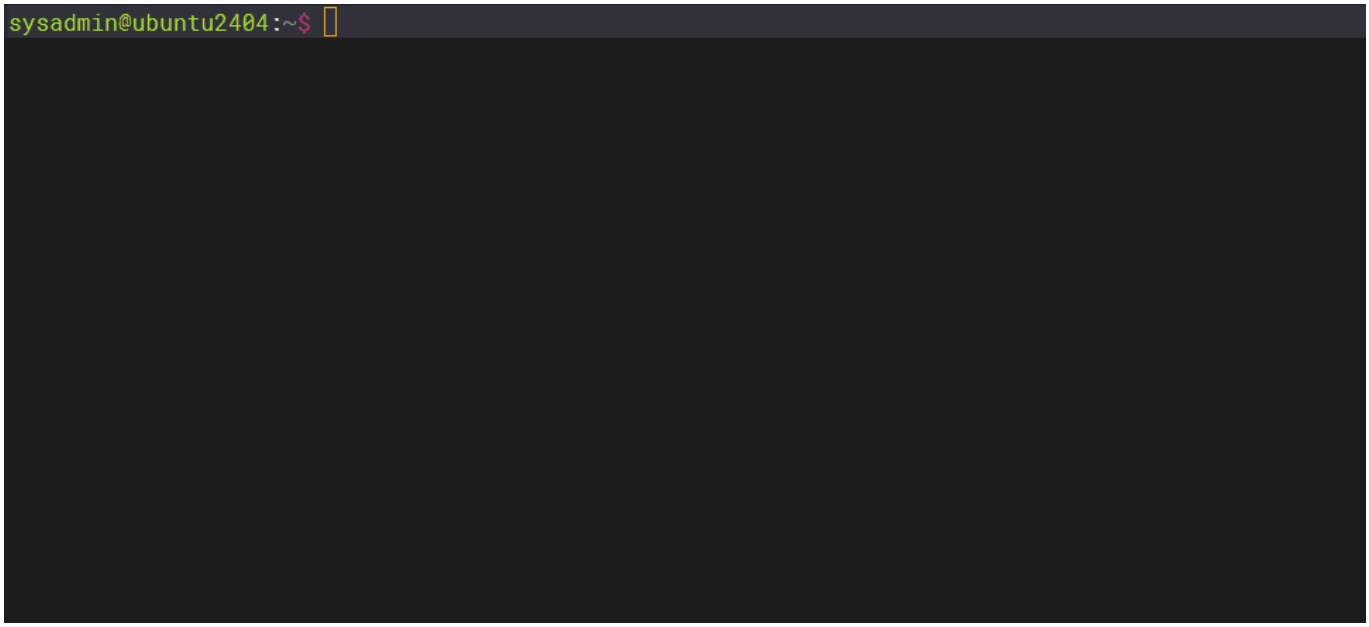
Change the port in Putty

If you can't access the Linux server, make sure you have opened the firewall on the server (you can open this page if you use RockyLinux and OpenSUSE, but if you use Ubuntu, you can read it on this page). If you want to access it via a Linux server, then use the format below:

```
ssh username@your_server_ip -p port_number
```

Then the format above can be the command below:

```
ssh sysadmin@192.168.56.12 -p 43210
```



Access via SSH using a new SSH port

WARNING

If you run a firewall on your remote server, you must open the port first. If you want to get an explanation of how to open the port, go to this page if you use firewalld or go to this page if you use ufw.

Note

Please note that the port number is from 0-65536, however, these ports are divided into 3 classifications:

- **Port 0-1023** => Well-Known ports, you can not use these ports.
- **Port 1024-49151** => Registered ports, these is a registered ports assigned by IANA (Internet Assigned Numbers Authority), you can or can not use these ports.
- **Port 49152-65535** => Dynamic or Private ports, you can use these ports.

References

jay75chauhan.medium.com
ionos.com

gcore.com
en.wikipedia.org
arubanetworks.com

How to Open and Close a Port in Ubuntu?

written by sysadmin | 5 February 2025

The previous article explained how to open and close ports in RockyLinux/AlmaLinux/CentOS. This article will explain how to open and close a port in Ubuntu.

Problem

How to open and close a port in Ubuntu?

Solution

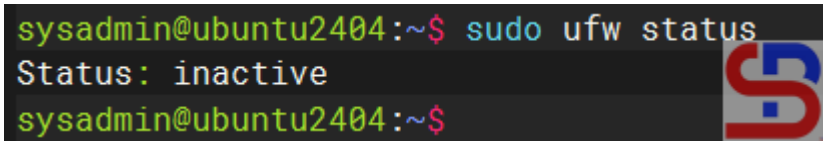
A. Check the firewall

By default, Ubuntu and Debian use the UFW or Uncomplicated Firewall application as the default firewall, and it is installed automatically when you install Ubuntu/Debian. If the firewall is not installed on your Ubuntu/Debian distro, use the command below:

```
sudo apt install ufw
```

To see whether ufw is running or not, use the command below:

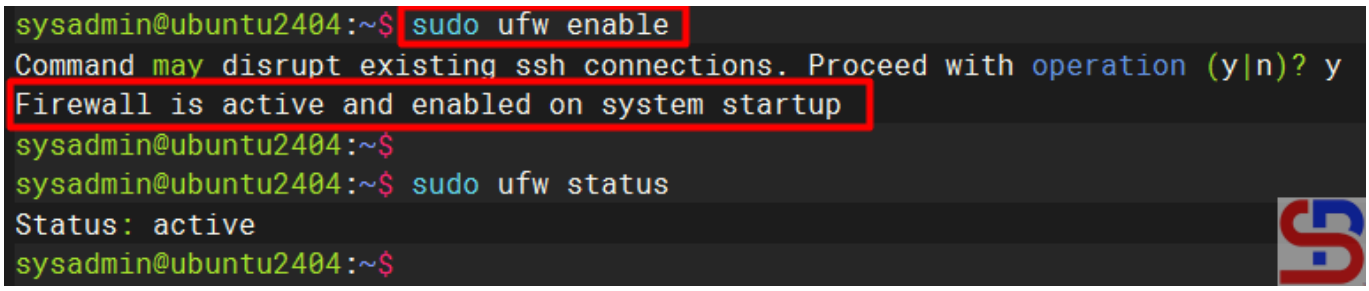
```
sudo ufw status
```



Check status ufw

From the image above, you can see that the application is not yet active. To enable it, type the command below:

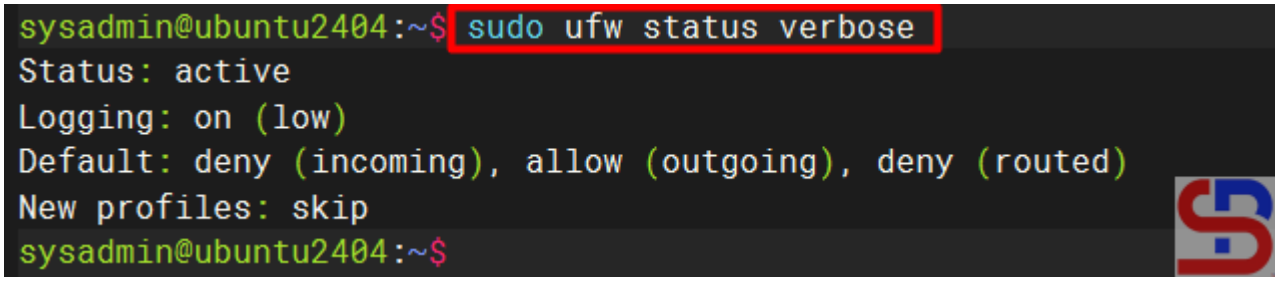
```
sudo ufw enable
```



Enable ufw

If you want to see the complete current status of the firewall, use the command below:

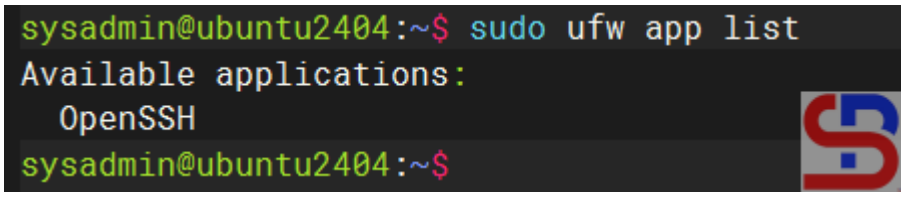
```
sudo ufw status verbose
```



Display the complete current status of the firewall

By default, the firewall only opens the OpenSSH service, which you can view by using the command below:

```
sudo ufw app list
```



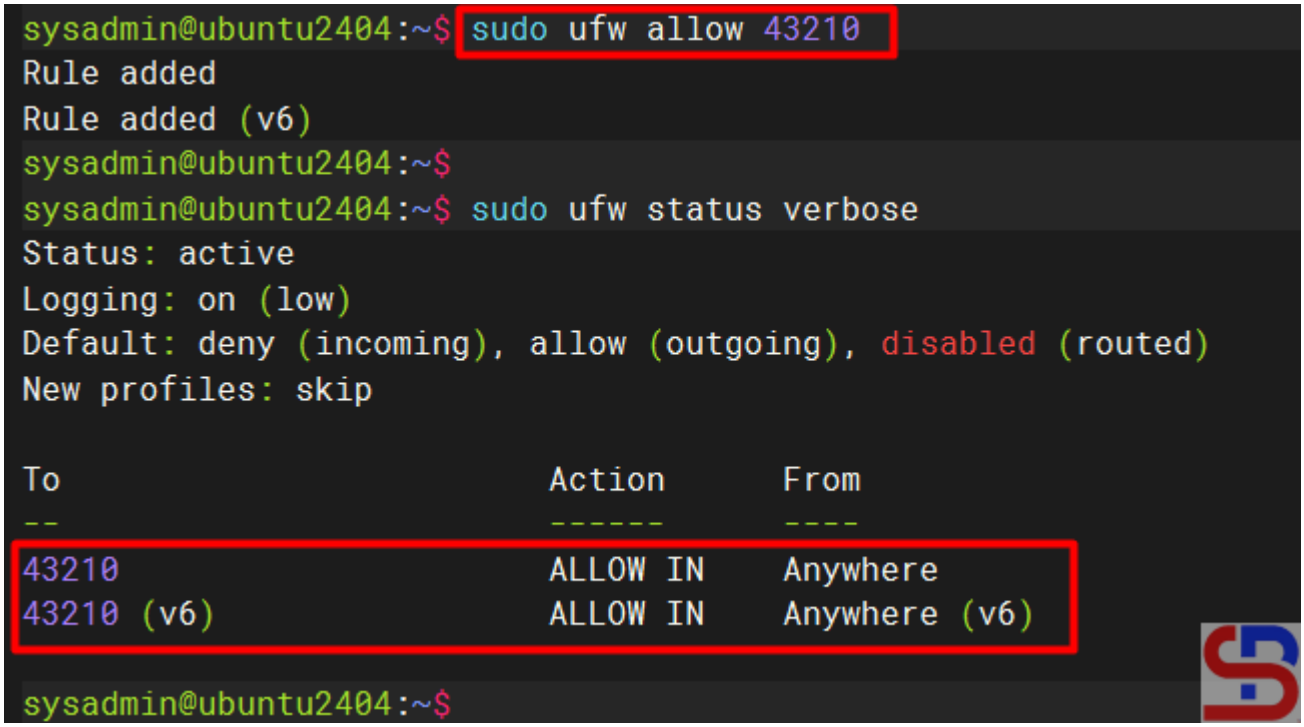
Display the service that is open in the firewall

B. Open the port

To open a port, for example, port 43210, use the command

below:

```
sudo ufw allow 43210
```



Open the port

WARNING
If you open the port using the command above, it means you will open the port for both TCP and UDP.

To open a port range, for example, from port numbers 45000 to 45010 with the TCP protocol, use the command below:

```
sudo ufw allow 45000:45010/tcp
```

```
sysadmin@ubuntu2404:~$ sudo ufw allow 45000:45010/tcp
Rule added
Rule added (v6)
sysadmin@ubuntu2404:~$
sysadmin@ubuntu2404:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
43210                      ALLOW       Anywhere
25/tcp                     ALLOW       Anywhere
22                         ALLOW       192.168.56.1
45000:45010/tcp            ALLOW       Anywhere
43210 (v6)                 ALLOW       Anywhere (v6)
25/tcp (v6)                ALLOW       Anywhere (v6)
45000:45010/tcp (v6)       ALLOW       Anywhere (v6)

sysadmin@ubuntu2404:~$
```

Open the range ports

C. Open the service

You can see from the image above that port 43210 has been opened on your Ubuntu server. You can also use the service name when opening a port. For example, if you want to open the SMTP service on your Ubuntu server, then use the command below:

```
sudo ufw allow smtp
```

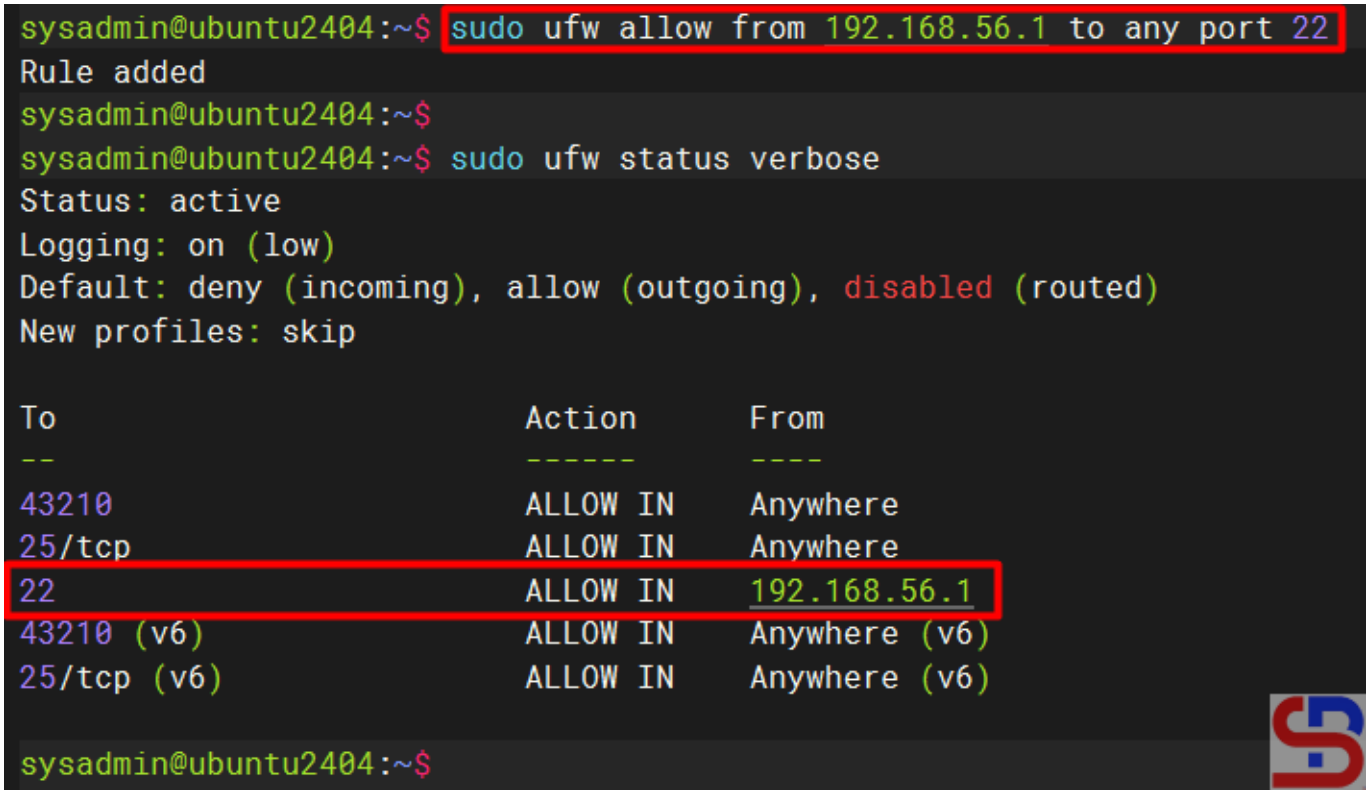
```
sysadmin@ubuntu2404:~$ sudo ufw allow smtp
Rule added
Rule added (v6)
sysadmin@ubuntu2404:~$
sysadmin@ubuntu2404:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
43210                      ALLOW       Anywhere
25/tcp                     ALLOW       Anywhere
43210 (v6)                 ALLOW       Anywhere (v6)
25/tcp (v6)                ALLOW       Anywhere (v6)

sysadmin@ubuntu2404:~$
```

Open the SMTP service

D. Open the port from a certain IP

If you want to open a port from a certain IP, for example, you only allow IP 192.168.56.1 to access port 22 on this server, then use the command below:

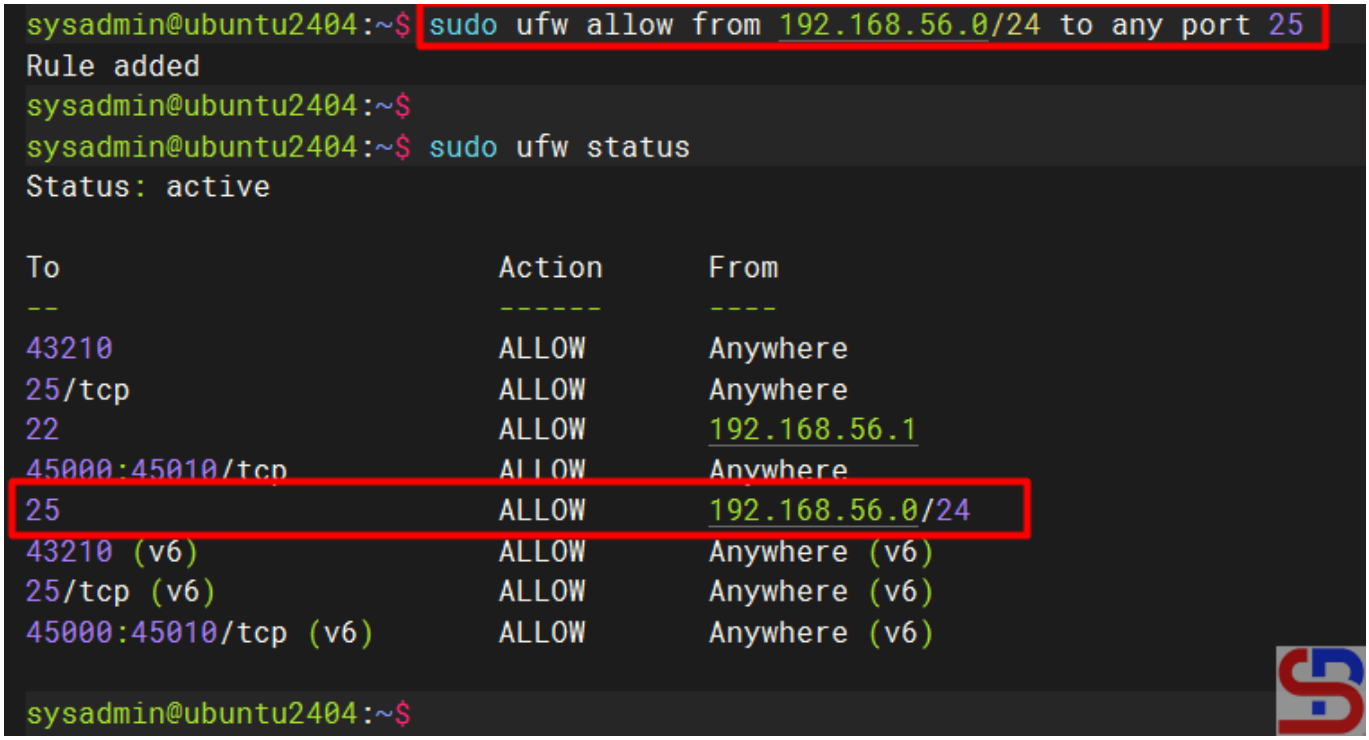
```
sudo ufw allow from 192.168.56.1 to any port 22
```



Allow the IP to a certain port

To allow the 192.168.56.0 subnet to the SMTP service, use the command below:

```
sudo ufw allow from 192.168.56.0/24 to any port 25
```



Allow the subnet to a certain port

E. Close the port

To close port 25, use the command below:

```
sudo ufw deny 25
```

```
sysadmin@ubuntu2404:~$ sudo ufw deny 25
Rule added
Rule added (v6)
sysadmin@ubuntu2404:~$
sysadmin@ubuntu2404:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
43210                      ALLOW       Anywhere
25/tcp                     ALLOW       Anywhere
22                         ALLOW       192.168.56.1
45000:45010/tcp            ALLOW       Anywhere
25                         ALLOW       192.168.56.0/24
25                         DENY        Anywhere
43210 (v6)                 ALLOW       Anywhere (v6)
25/tcp (v6)                ALLOW       Anywhere (v6)
45000:45010/tcp (v6)       ALLOW       Anywhere (v6)
25 (v6)                    DENY        Anywhere (v6)

sysadmin@ubuntu2404:~$
```

Close the port

F. Delete the port

You can also close a port and delete the port that has been opened, for example, port 43210, using the syntax below:

```
sudo ufw delete number
```

```
sysadmin@ubuntu2404:~$ sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 43210                      ALLOW IN    Anywhere
[ 2] 25/tcp                     ALLOW IN    Anywhere
[ 3] 22                         ALLOW IN    192.168.56.1
[ 4] 45000:45010/tcp            ALLOW IN    Anywhere
[ 5] 25                         ALLOW IN    192.168.56.0/24
[ 6] 25                         DENY IN     Anywhere
[ 7] 43210 (v6)                 ALLOW IN    Anywhere (v6)
[ 8] 25/tcp (v6)                ALLOW IN    Anywhere (v6)
[ 9] 45000:45010/tcp (v6)       ALLOW IN    Anywhere (v6)
[10] 25 (v6)                    DENY IN     Anywhere (v6)

sysadmin@ubuntu2404:~$ sudo ufw delete 1
Deleting:
 allow 43210
Proceed with operation (y|n)? y
Rule deleted
sysadmin@ubuntu2404:~$
```

Close and delete the port

WARNING

You don’t need to run **sudo ufw reload** after each rule change using ufw commands (such as ufw allow or ufw deny). However, you will need to run **sudo ufw reload** if you are editing the ufw configuration file manually (such as /etc/ufw/before.rules or /etc/ufw/after.rules), or if you want to make sure all the latest rules and settings are loaded.

Note

You can remove all the rules in ufw by using the command below:

sudo ufw reset

After that, enable the ufw by using the command below:

sudo ufw enable

```
sysadmin@Ubuntu2404:~$ sudo ufw reset
Resetting all rules to installed defaults. This may disrupt existing ssh
connections. Proceed with operation (y|n)? y
Backing up 'user.rules' to '/etc/ufw/user.rules.20250515_081802'
Backing up 'before.rules' to '/etc/ufw/before.rules.20250515_081802'
Backing up 'after.rules' to '/etc/ufw/after.rules.20250515_081802'
Backing up 'user6.rules' to '/etc/ufw/user6.rules.20250515_081802'
Backing up 'before6.rules' to '/etc/ufw/before6.rules.20250515_081802'
Backing up 'after6.rules' to '/etc/ufw/after6.rules.20250515_081802'

sysadmin@Ubuntu2404:~$ sudo ufw status
Status: inactive
sysadmin@Ubuntu2404:~$ sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
sysadmin@Ubuntu2404:~$
```

Reset ufw

By default, if you open a port, it will automatically open in IPv4 and IPv6, and likewise, if you close the port. To see the UFW settings, open the **/etc/default/ufw** file.

```
sysadmin@ubuntu2404:~$ cat /etc/default/ufw
# /etc/default/ufw
#

# Set to yes to apply rules to support IPv6 (no means only IPv6 on loopback
# accepted). You will need to 'disable' and then 'enable' the firewall for
# the changes to take affect.
IPV6=yes

# Set the default input policy to ACCEPT, DROP, or REJECT. Please note that if
# you change this you will most likely want to adjust your rules.
DEFAULT_INPUT_POLICY="DROP"

# Set the default output policy to ACCEPT, DROP, or REJECT. Please note that if
# you change this you will most likely want to adjust your rules.
DEFAULT_OUTPUT_POLICY="ACCEPT"

# Set the default forward policy to ACCEPT, DROP or REJECT.  Please note that
# if you change this you will most likely want to adjust your rules
DEFAULT_FORWARD_POLICY="DROP"

# Set the default application policy to ACCEPT, DROP, REJECT or SKIP. Please
# note that setting this to ACCEPT may be a security risk. See 'man ufw' for
# details
DEFAULT_APPLICATION_POLICY="SKIP"
```

Configuration of ufw

References

cyberciti.biz
phoenixnap.com
digitalocean.com
help.ubuntu.com
askubuntu.com

How to Install gcloud on Ubuntu?

written by sysadmin | 5 February 2025

The previous article explained how the install gcloud on RockyLinux/AlmaLinux/CentOS. This article will explain how to install gcloud on Ubuntu.

Problem

How to install gcloud on Ubuntu?

Solution

Here are the steps to install gcloud on Ubuntu/Debian:

A. Install gcloud

As far as I know, there are 3 methods to install gcloud on Ubuntu/Debian and the methods recommend using a user other than root.

1. Using the script

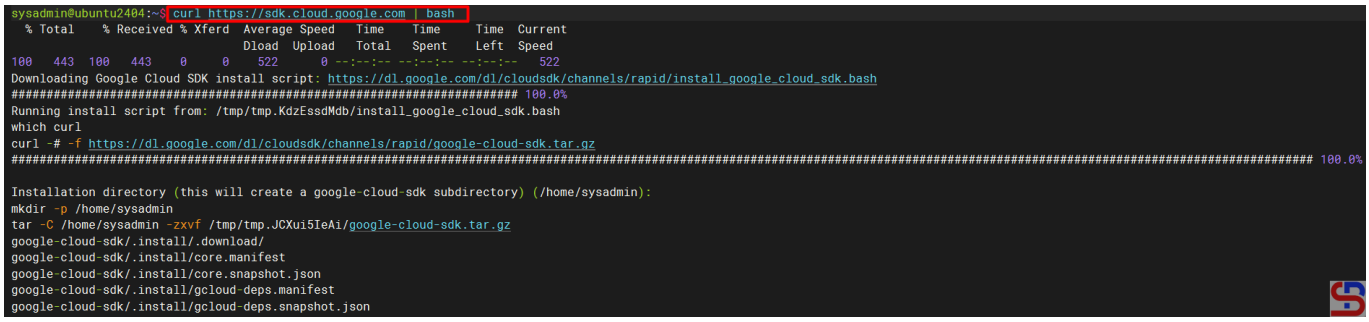
Before you download the script, install the packages using the command below:

```
sudo apt update
sudo apt-get install curl tar
```

Use the below command to download and install the script:

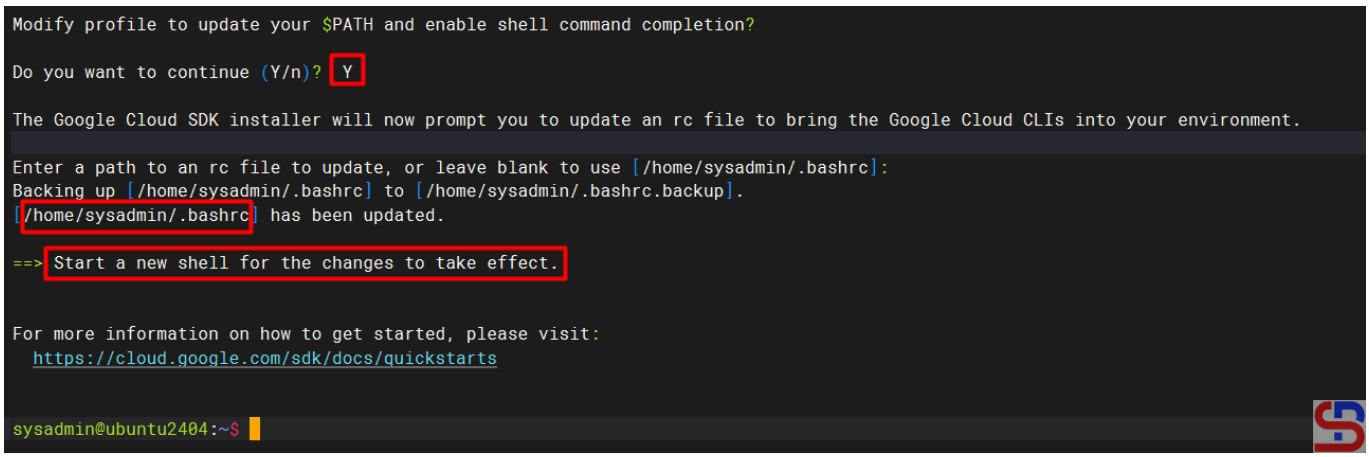
```
curl https://sdk.cloud.google.com | bash
```

Then you will see a display like the one below:



Install gcloud using the script

Wait until it’s finished, and you will see a display like the one below:



Installation complete

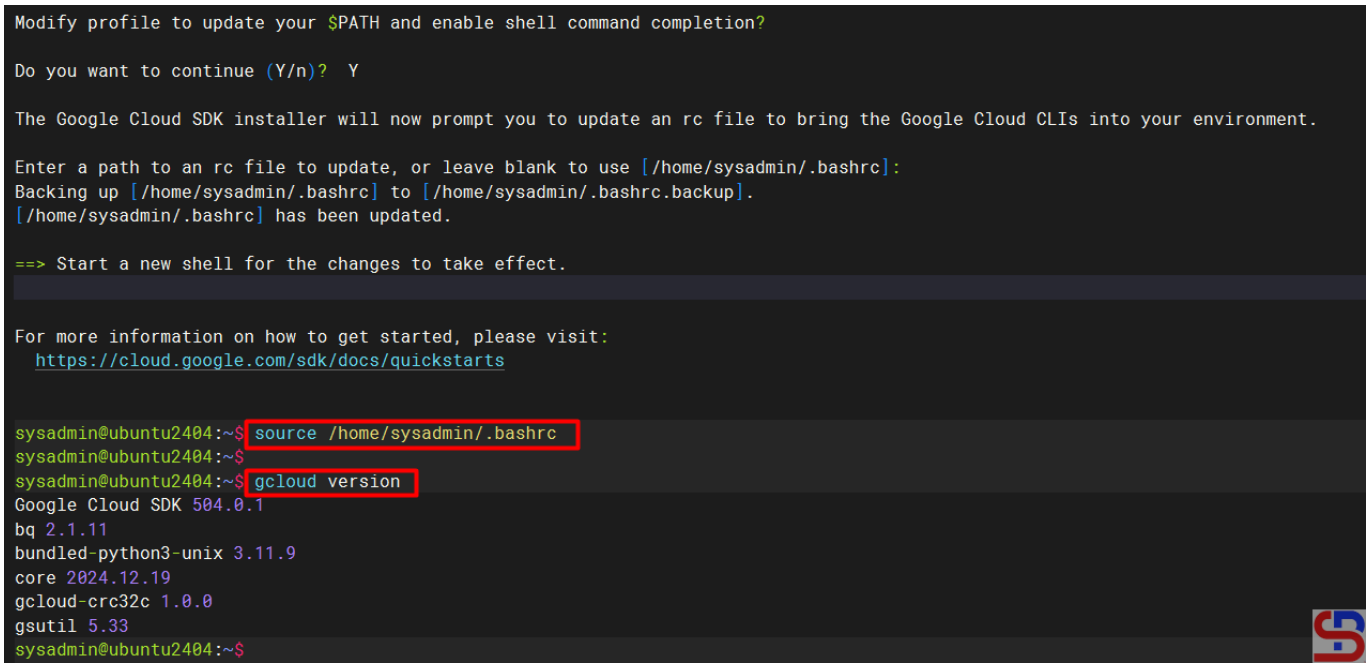
From the image above, you are asked to create a new SSH connection so that the effect can be seen, and type the command below:

```
gcloud version
```

However, you can use the command below:

```
source /home/sysadmin/.bashrc
```

So you don’t need to create a new SSH connection to run the gcloud version command, which results in the image below:



Check the result of the installation

2. Using the repository

Type the following commands to install gcloud on the Ubuntu/Debian distro:

```
sudo apt update
echo 'deb [signed-by=/usr/share/keyrings/cloud.google.gpg]
https://packages.cloud.google.com/apt cloud-sdk main' | sudo tee -a
sudo apt-get -y install apt-transport-https ca-certificates gnupg
curl https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key --
keyring /usr/share/keyrings/cloud.google.gpg add -
sudo apt update
sudo apt-get install -y google-cloud-sdk
```

3. Using the snap

Run the below command to install gcloud:

```
sudo snap install google-cloud-sdk --classic
```

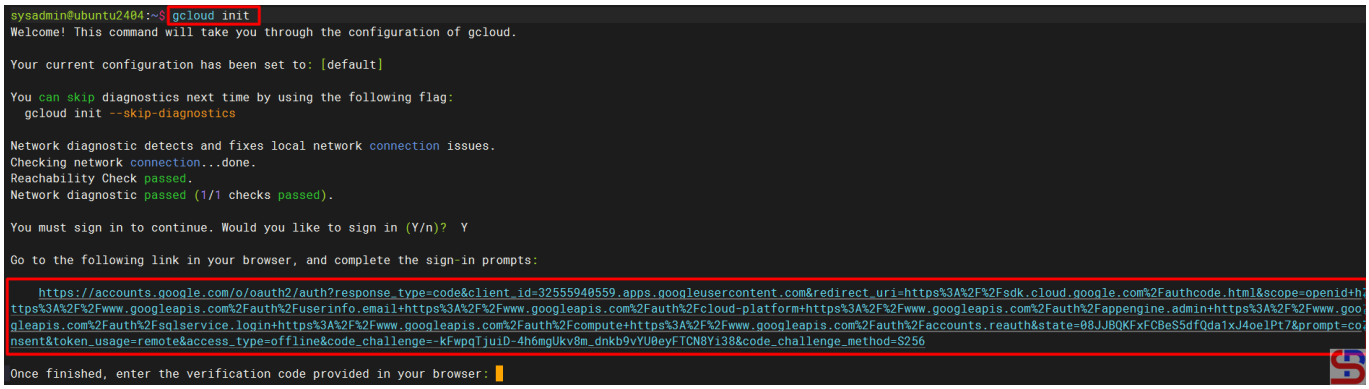
B. Connect to GCP

After you install gcloud on your server, type the command

below:

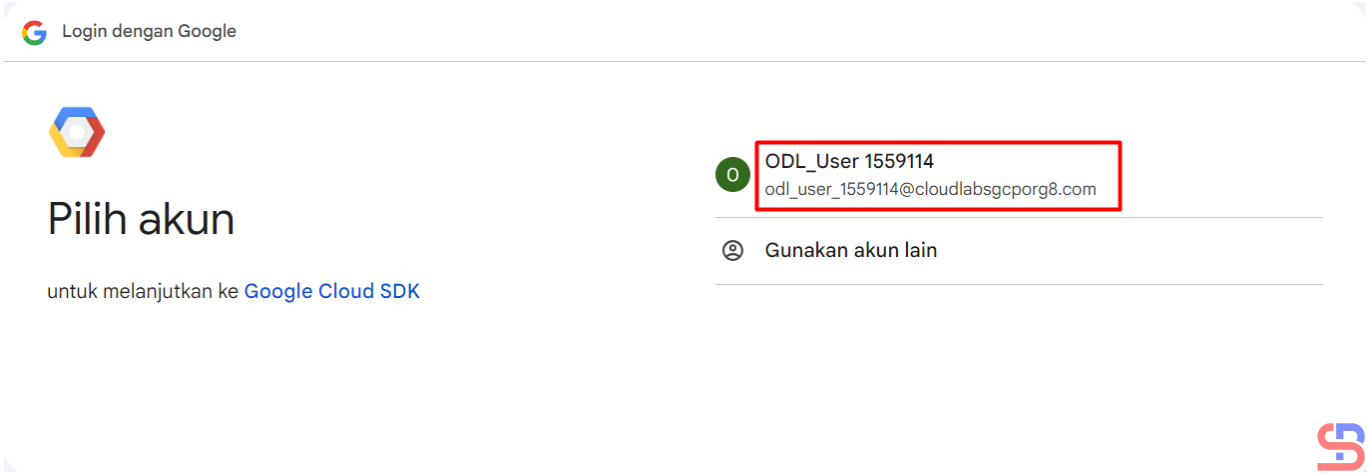
```
gcloud init
```

Then there will be a display like the image below:



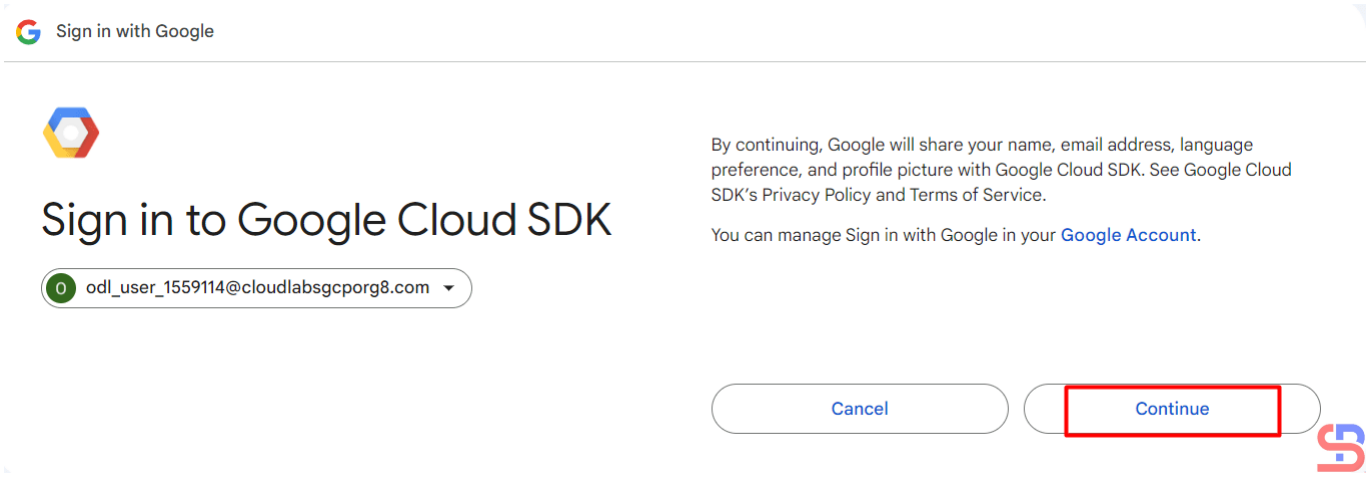
Click the link

You can open the link in a browser by clicking the **Ctrl+Click** button located in the red box. If you are having trouble doing so, copy what is included in the red box and paste it into your browser. This will allow you to view a display similar to the one that is shown below:



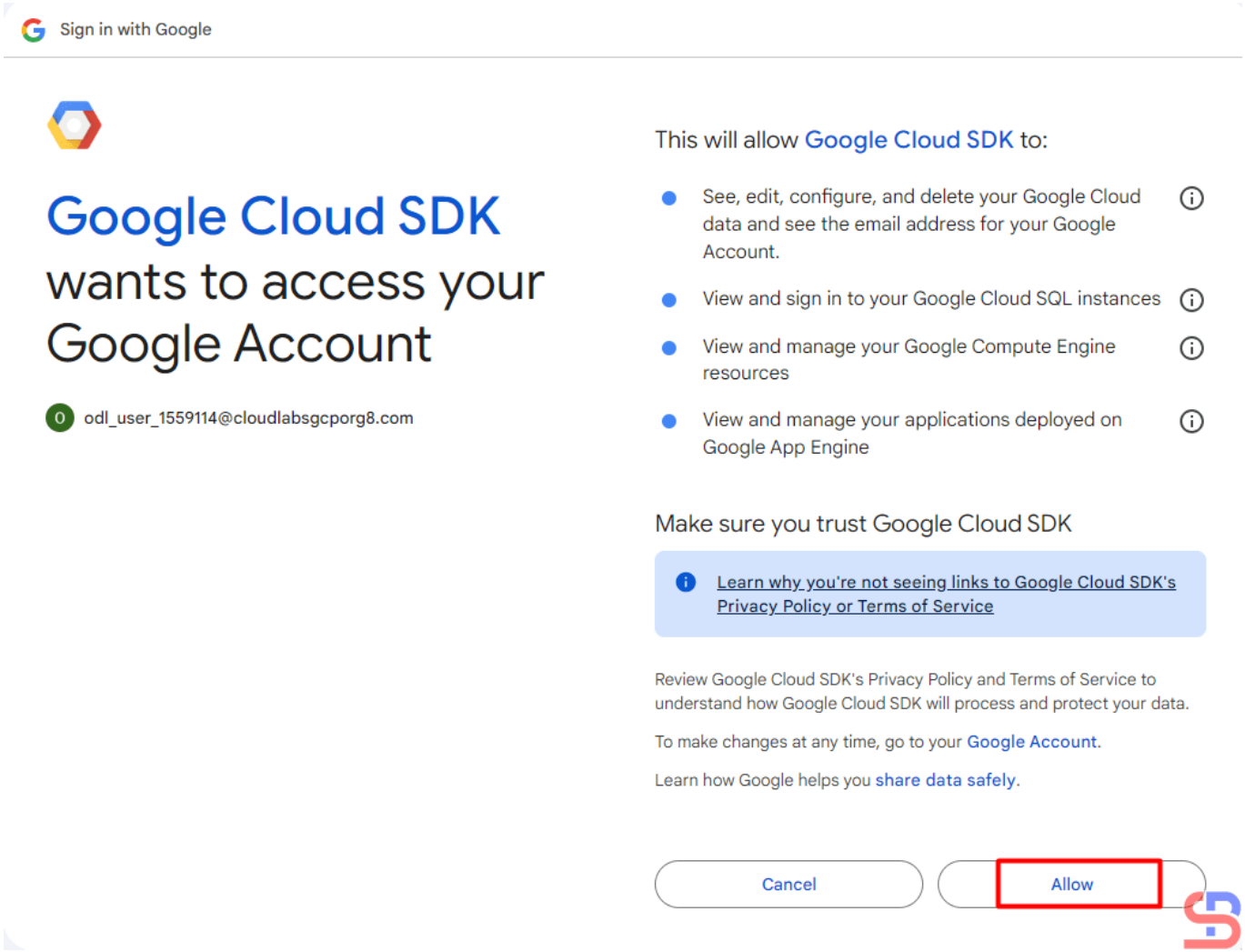
Click the account

When you click on your Google account, that will allow you to access GCP, and a display similar to the one shown below will appear:



Click the Continue button

After you click the **Continue** button, the screen below will show:



Click the Allow button

When you click the **Allow** button, the screen below will show:

Sign in to the gcloud CLI

You are seeing this page because you ran the following command in the gcloud CLI from this or another machine. If this is not the case, close this tab.

```
gcloud auth login --no-launch-browser
```

Enter the following verification code in gcloud CLI on the machine you want to log into. This is a credential **similar to your password** and should not be shared with others.

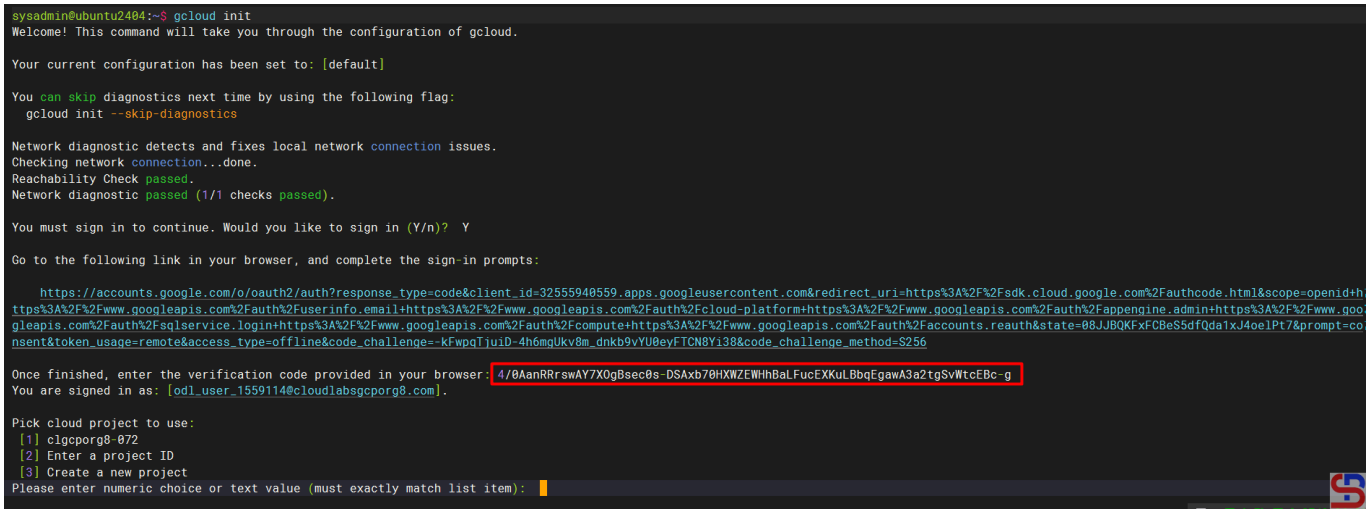
```
4/0AanRRrswAY7XOgBsec0s-DSAxb70HXWZEWH
hBaLFucEXKuLBbqEgawA3a2tgSvWtcEBc-g
```

Copy

You can close this tab when you're done.

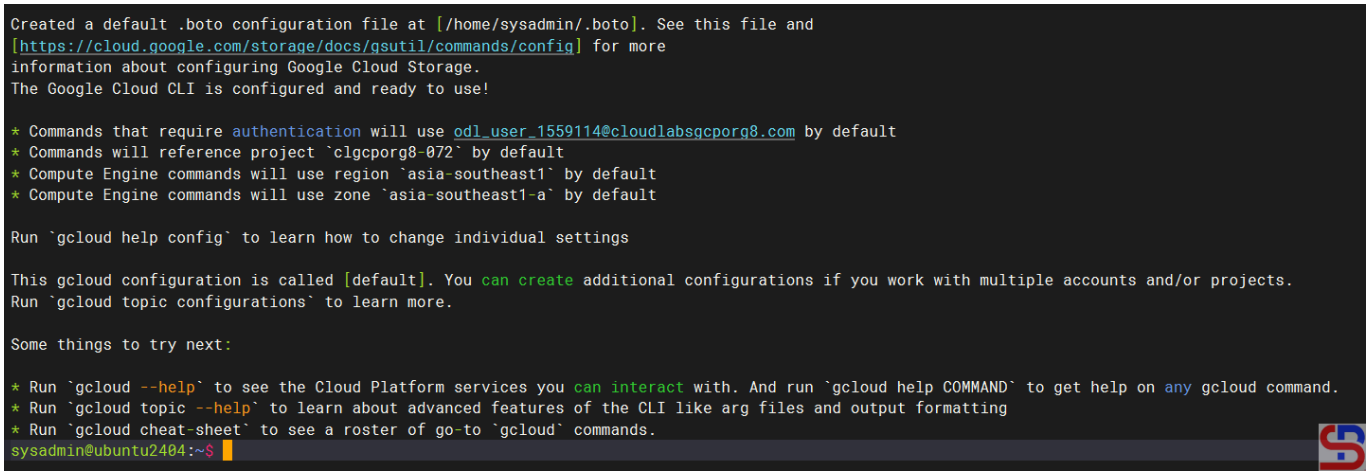
Click the Copy button

Click the **Copy** button, and paste it into the CLI on your server as in the image below:



Paste the code

Select the project and configure the zone as in the image above. After that, the gcloud configuration is complete, like in the image below:



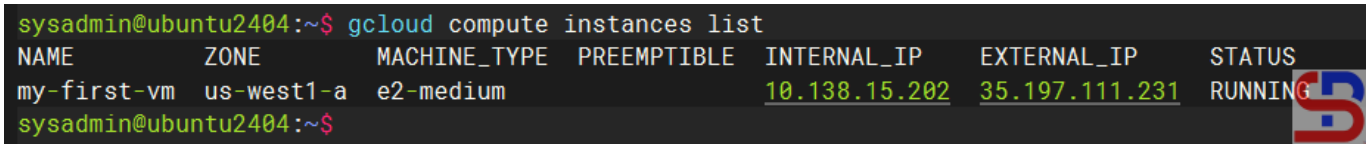
Installation of GCP is complete

C. Test gcloud

Now, try gcloud to access your GCP. I try to list my virtual machine in GCP using the below command:

```
gcloud compute instances list
```

Then the display below will appear:



Display virtual machine in GCP using gcloud

If you get a display like the image above, then you have successfully used your GCloud to access your GCP.

Note

If you have many projects on your GCP, you can choose one of these projects as the starting point for your gcloud on GCP. You can switch projects using the command:

```
gcloud config set project PROJECT_ID
```

Change **PROJECT_ID** to the project ID you want to switch to.

References

cloud.google.com
liquidweb.com
bacancytechnology.com
attuneops.io
tecadmin.net

How to Open And Close a Port on RockyLinux Server?

written by sysadmin | 5 February 2025

By default, the RockyLinux/AlmaLinux/CentOS distro provides two firewalls, iptables and firewalld. This article will explain how to open and close a port using Firewalld on the distro. If you have opened and closed a port using Firewalld, you don’t need to open and close a port in iptables.

Problem

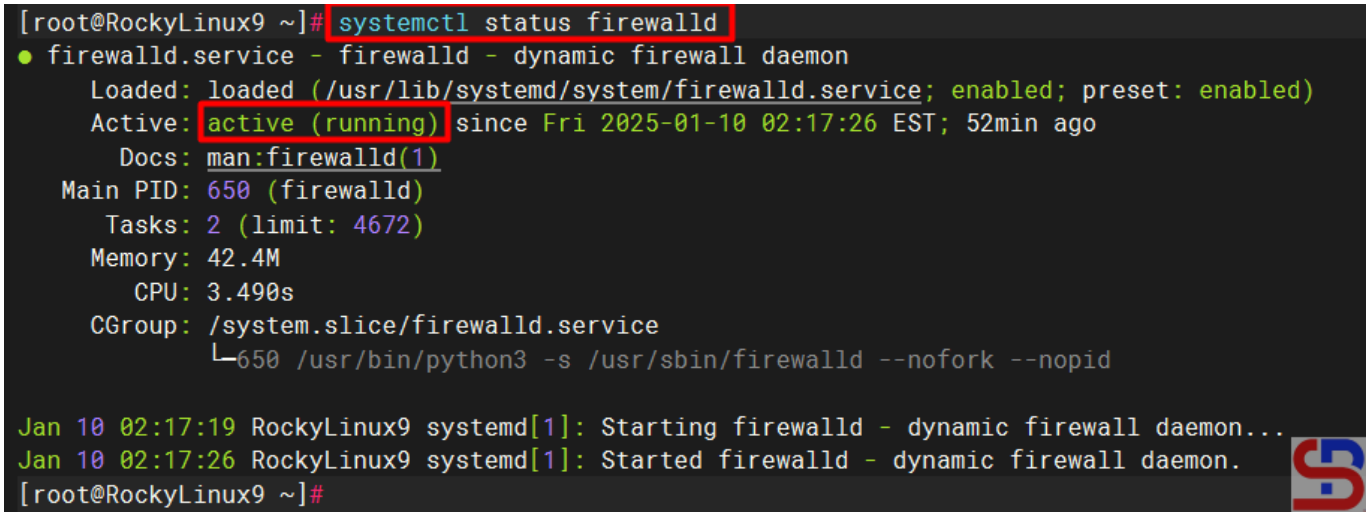
How to open and close a port on the RockyLinux server?

Solution

A. Check the Firewalld status

By default, the Firewalld package is installed automatically using the command:

```
systemctl status firewalld
```



Check the status of Firewalld

From the picture above, you can see that the firewall on the server is already running. If the Firewalld is not already running, use the command below:

```
systemctl enable --now firewalld
```

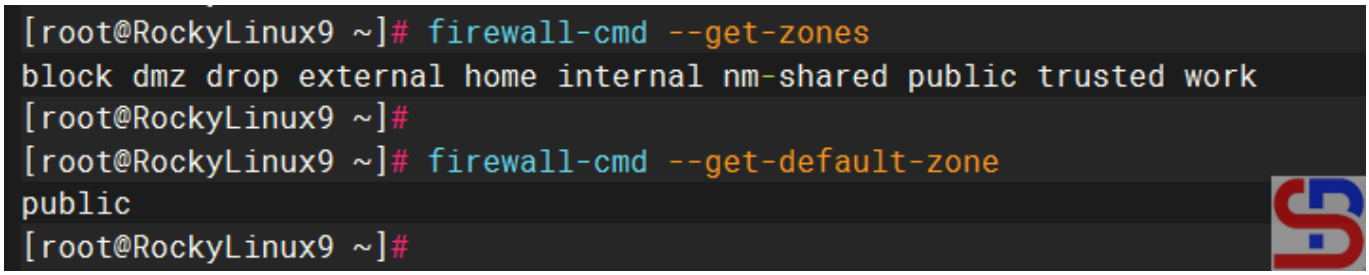
But if on your server there is no firewall package, you can install it using the command below:

```
yum install -y firewalld
```

B. Check the zones

Firewalld uses zones and services, compared to iptables, which use chains and rules. Zones are a collection of rules that have been set for what network connections should be permitted based on the level of confidence in the network connected to the system. We can determine the name of the network interface and the network source into zones. To see the zones in firewalld and which zone is the default, use the command below:

```
firewall-cmd --get-zones
firewall-cmd --get-default-zone
```



Show all zones in Firewalld

From the picture above, there are 9 zones, and the explanation can be seen in the picture below, which is sorted from the most trusted

Zone Name	Description
Trusted	This zone accepts all the incoming traffic. You can use this zone to manage the traffic on a trusted network because it will not filter anything.
Home	This zone is designed for only the home network. It permits only selected incoming traffic and reject all.
Work	This zone designed for only the work (corporate) networking. It permits only selected incoming traffic and reject all.
Internal	This zone intended to design for the internal network. It permits only what is allowed and rejects all.
Public	This zone rejects all the incoming traffic, except what is granted. Using with the default zone, we can add any newly network interfaces on it. It is designed to use only the public places.
External	This zone designed for outgoing traffic forwarded with masquerading is enabled. Also, we can use this for NAT
Dmz	This zone designed to use the demilitarized zone with limited public access. It permits only selected incoming traffic and reject all.
Block	This zone designed to reject all incoming traffic with an ICMP-host-prohibited message is returned. It permits only outgoing traffic.
Drop	This zone designed to drop all incoming traffic with no notification like ICMP errors. It is purely used in high secure places.

The zones in Firewalld (Image credit for linuxteck.com)

To view all settings for all zones, use the following command:

```
firewall-cmd --list-all-zones
```

```
[root@RockyLinux9 ~]# firewall-cmd --list-all-zones
block
  target: %%REJECT%%
  icmp-block-inversion: no
  interfaces:
  sources:
  services:
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:

dmz
  target: default
  icmp-block-inversion: no
  interfaces:
  sources:
  services: ssh
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

View all the settings in Firewalld

But, if you want to view all settings in a specific zone, for example, a public zone, use the following command:

```
firewall-cmd --zone=public --list-ports
```

C. Open the Port

Now, if you want to open port 43210 with TCP protocol, use the command below:

```
firewall-cmd --add-port=43210/tcp --permanent
firewall-cmd --reload
```

```
[root@RockyLinux9 ~]# firewall-cmd --add-port=43210/tcp --permanent
success
[root@RockyLinux9 ~]#
[root@RockyLinux9 ~]# firewall-cmd --reload
success
[root@RockyLinux9 ~]#
```

Open the port

Use the command below to see the ports that have been opened:

```
firewall-cmd --list-ports
```

```
[root@RockyLinux9 ~]# firewall-cmd --list-ports
43210/tcp
[root@RockyLinux9 ~]#
```

List all opened ports

D. Open the port from a certain IP

If you want to open a port from a certain IP, for example, you only allow IP 192.168.56.100 to access port 22 on this server, then use the command below:

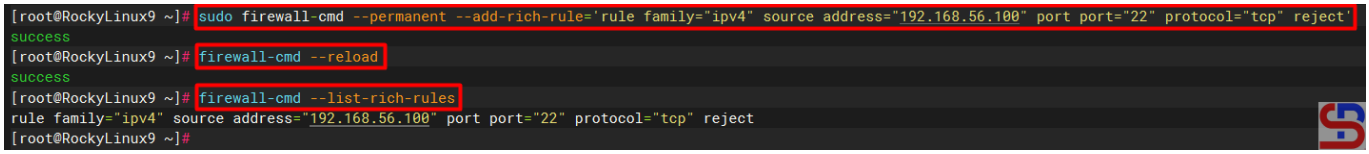
```
firewall-cmd --zone=public --add-rich-rule 'rule family=ipv4 source
address=192.168.56.100 port port=22 protocol=tcp accept'
firewall-cmd --reload
firewall-cmd --list-rich-rules
```

```
[root@RockyLinux9 ~]# sudo firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="192.168.56.100" port port="22" protocol="tcp" accept'
success
[root@RockyLinux9 ~]# firewall-cmd --reload
success
[root@RockyLinux9 ~]# firewall-cmd --list-rich-rules
rule family="ipv4" source address="192.168.56.100" port port="22" protocol="tcp" accept
[root@RockyLinux9 ~]#
```

Allow the IP to a certain port

If you want to reject a host with IP 192.168.56.100 to access port 22, use the command below:

```
sudo firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="192.168.56.100" port port="22" protocol="tcp" reject'
firewall-cmd --reload
firewall-cmd --list-rich-rules
```

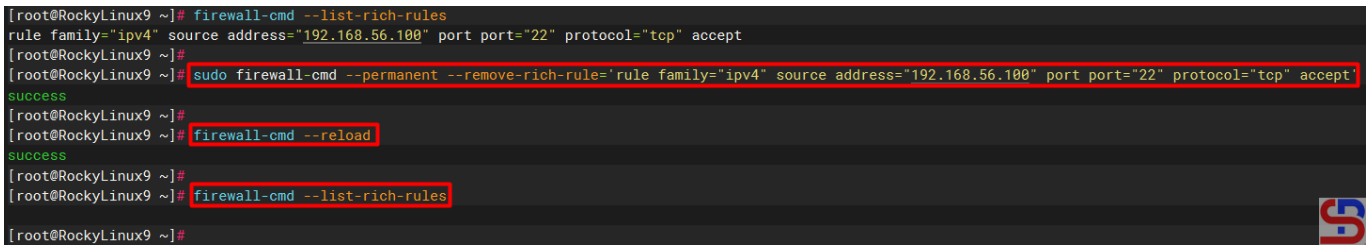


Block the IP to a certain port

E. Close the port from a certain IP

If you want to close a port from a certain IP, for example, you block a host with IP 192.168.56.100 from accessing port 22 on this server, then use the command below:

```
sudo firewall-cmd --permanent --remove-rich-rule='rule family="ipv4" source address="192.168.56.100" port port="22" protocol="tcp" accept'
firewall-cmd --reload
firewall-cmd --list-rich-rules
```



Remove the IP to a certain port

INFO

In short, if you want to delete the rich rule, then change the option –add-rich-rule to **–remove-rich-rule**.

F. Close the port

Use the command below to close the newly opened port 43210:

```
firewall-cmd --remove-port=43210/tcp --permanent
firewall-cmd --reload
```

```
[root@RockyLinux9 ~]# firewall-cmd --list-ports
43210/tcp
[root@RockyLinux9 ~]#
[root@RockyLinux9 ~]# firewall-cmd --remove-port=43210/tcp --permanent
success
[root@RockyLinux9 ~]#
[root@RockyLinux9 ~]# firewall-cmd --reload
success
[root@RockyLinux9 ~]#
[root@RockyLinux9 ~]# firewall-cmd --list-ports

[root@RockyLinux9 ~]#
```

Close the port in Firewalld

G. Open the service

Apart from using ports, Firewalld can also open and close services on the server. To see the services that have been opened, type the command below:

```
firewall-cmd --list-services
```

```
[root@RockyLinux9 ~]# firewall-cmd --list-services
cockpit dhcpv6-client ssh
[root@RockyLinux9 ~]#
```

List all opened services

You can see in the picture above that the distro only opens 3 services. If you want to open the SMTP service, use the command below:

```
firewall-cmd --add-service=smtp --permanent
firewall-cmd --reload
```

```
[root@RockyLinux9 ~]# firewall-cmd --add-service=smtp --permanent
success
[root@RockyLinux9 ~]#
[root@RockyLinux9 ~]# firewall-cmd --reload
success
[root@RockyLinux9 ~]#
[root@RockyLinux9 ~]# firewall-cmd --list-services
cockpit dhcpv6-client smtp ssh
[root@RockyLinux9 ~]#
```

Add the service to the firewall

H. Close the service

To delete the SMTP service in Firewalld, use the command below:

```
firewall-cmd --remove-service=smtp --permanent
firewall-cmd --reload
```

Close the service in Firewalld

Note

If you use the OpenSUSE distro, you can use the above commands to open and close a port, like in the image below:

```
opensuse15:~ # systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
     Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; preset: disabled)
     Active: active (running) since Fri 2025-01-10 06:14:05 EST; 5min ago
       Docs: man:firewalld(1)
   Main PID: 833 (firewalld)
      Tasks: 2 (limit: 1125)
        CPU: 23.153s
     CGroup: /system.slice/firewalld.service
             └─833 /usr/bin/python3 /usr/sbin/firewalld --nofork --nopid

Jan 10 06:13:57 opensuse15 systemd[1]: Starting firewalld - dynamic firewall daemon...
Jan 10 06:14:05 opensuse15 systemd[1]: Started firewalld - dynamic firewall daemon.
opensuse15:~ #
opensuse15:~ # firewall-cmd --add-port=43210/tcp --permanent
success
opensuse15:~ #
opensuse15:~ # firewall-cmd --reload
success
opensuse15:~ #
opensuse15:~ # firewall-cmd --list-ports
43210/tcp
opensuse15:~ #
opensuse15:~ # firewall-cmd --remove-port=43210/tcp --permanent
success
opensuse15:~ #
opensuse15:~ # firewall-cmd --reload
success
opensuse15:~ #
opensuse15:~ # firewall-cmd --list-ports

opensuse15:~ #
```

The Firewalld commands in OpenSUSE

References

redhat.com
greenwebpage.com
inmotionhosting.com
baeldung.com
musaamin.web.id