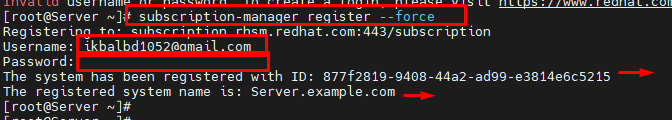
**Red Hat subscription-manager**

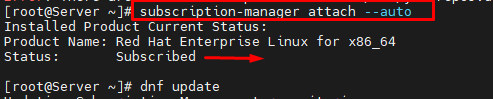
**[root@Server ~]# subscription-manager register –force**

Username: [**ikbalbd1052@gmail.com**](mailto:ikbalbd1052@gmail.com)

Password: Ikbal123456



**[root@Server ~]# subscription-manager attach –auto**



**SSH Daemon Configuration.**

**[root@Server ~]# useradd israt**

**[root@Server ~]# passwd israt**

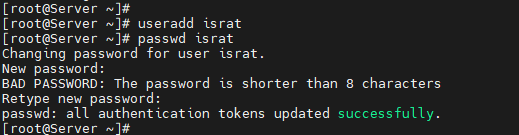
Changing password for user israt.

New password: 1234

BAD PASSWORD: The password is shorter than 8 characters

Retype new password:1234

passwd: all authentication tokens updated successfully.



[root@Server ~]# vim /etc/ssh/sshd\_config

/////Change The File.

Port 9722

PermitRootLogin no

AllowUsers israt ikbal

[root@Server ~]# visudo

## Allow root to run any commands anywhere

root ALL=(ALL) ALL

%israt ALL=(ALL) ALL, !/bin/su

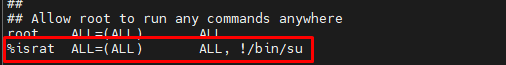
%ikbal ALL=(ALL) NOPASSWD: ALL /root login no password

**// Ubuntu Server**

# User privilege specification

root ALL=(ALL:ALL) ALL

ikbal ALL=(ALL:ALL) NOPASSWD: ALL



[root@Server ~]# semanage port -a -t ssh\_port\_t -p tcp 9722

[root@Server ~]# semanage port -m -t ssh\_port\_t -p tcp 9722

[root@Server ~]# systemctl restart sshd

[root@Server ~]# sudo firewall-cmd --add-port=9722/tcp --permanent

success

[root@Server ~]# sudo firewall-cmd --remove-service=ssh --permanent

success

[root@Server ~]# systemctl restart sshd.service

[root@Server ~]# firewall-cmd –state

[root@Server ~]# firewall-cmd --permanent --add-port=9722/tcp

[root@Server ~]# firewall-cmd --reload

root@ubuntu:~# apt install firewalld -y

root@ubuntu:~# systemctl start firewalld

root@ubuntu:~# systemctl enable firewalld

root@ubuntu:~# firewall-cmd --list-all

root@ubuntu:~# firewall-cmd --remove-port=9722/tcp

**Zimbra Mail Server Install Centos OS**

**Copy and Paste** vim /etc/redhat-release file and past centos vim /etc/redhat-release File A

[root@mail zcs-8.8.15\_GA\_3953.RHEL8\_64.20200629025823]# vim /etc/redhat-release



Red Hat Enterprise Linux release 8.7 (Ootpa)

User : admin@triangle.com.bd

Pass: !kbal@0171#

[ikbal@triangle.com.bd](mailto:ikbal@triangle.com.bd)

[root@tsl-mx1 ~]# grep mehedy.hasan /var/log/maillog | wc -L

401 //Mail Count Command

[root@zimbra zcs-8.8.15\_GA\_3953.RHEL8\_64.20200629025823]# systemctl start firewalld

[root@zimbra zcs-8.8.15\_GA\_3953.RHEL8\_64.20200629025823]# systemctl enable firewalld

[root@zimbra zcs-8.8.15\_GA\_3953.RHEL8\_64.20200629025823]# firewall-cmd --state

[zimbra@mail ~]$ /opt/zimbra/common/sbin/postqueue -p

[zimbra@mail ~]$ /opt/zimbra/common/sbin/postqueue -f

[zimbra@mail ~]$ su - zimbra

**Fix Failed to download metadata for repo-Yum**

[root@autocontroller ~]# yum update

CentOS-8 - AppStream 70 B/s | 38 B 00:00

Error: Failed to download metadata for repo 'AppStream': Cannot prepare internal mirrorlist: No URLs in mirrorlist

[root@autocontroller ~]# ping google.com

PING google.com (172.217.166.206) 56(84) bytes of data.

64 bytes from del03s13-in-f14.1e100.net (172.217.166.206): icmp\_seq=1 ttl=115 ti me=43.5 ms

--- google.com ping statistics ---

1 packets transmitted, 1 received, 0% packet loss, time 0ms

rtt min/avg/max/mdev = 43.508/43.508/43.508/0.000 ms

**Step 1:** Go to the /etc/yum.repos.d/ directory.

[root@autocontroller ~]# cd /etc/yum.repos.d/

**Step 2:** Run the below commands

[root@autocontroller ~]# sed -i 's/mirrorlist/#mirrorlist/g' /etc/yum.repos.d/CentOS-\*

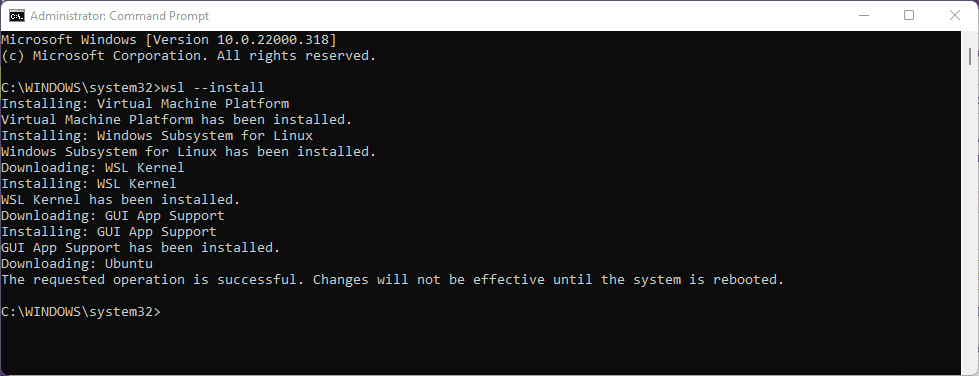
[root@autocontroller ~]# sed -i 's|#baseurl=http://mirror.centos.org|baseurl=http://vault.centos.org|g' /etc/yum.repos.d/CentOS-\*

**Step 3:** Now run the yum update

[root@autocontroller ~]# yum update -y

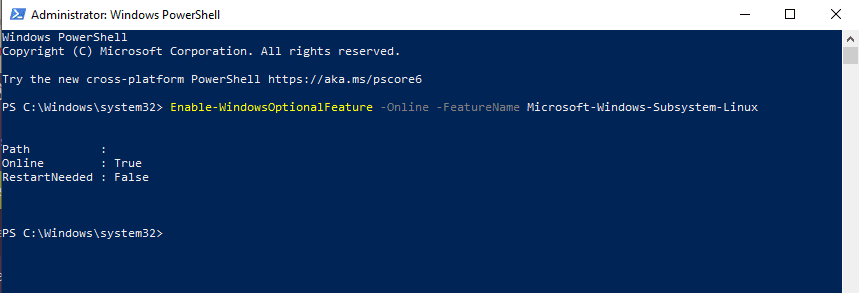
**Install CentOS using WSL**

wsl --install



Step 1: Enable WSL

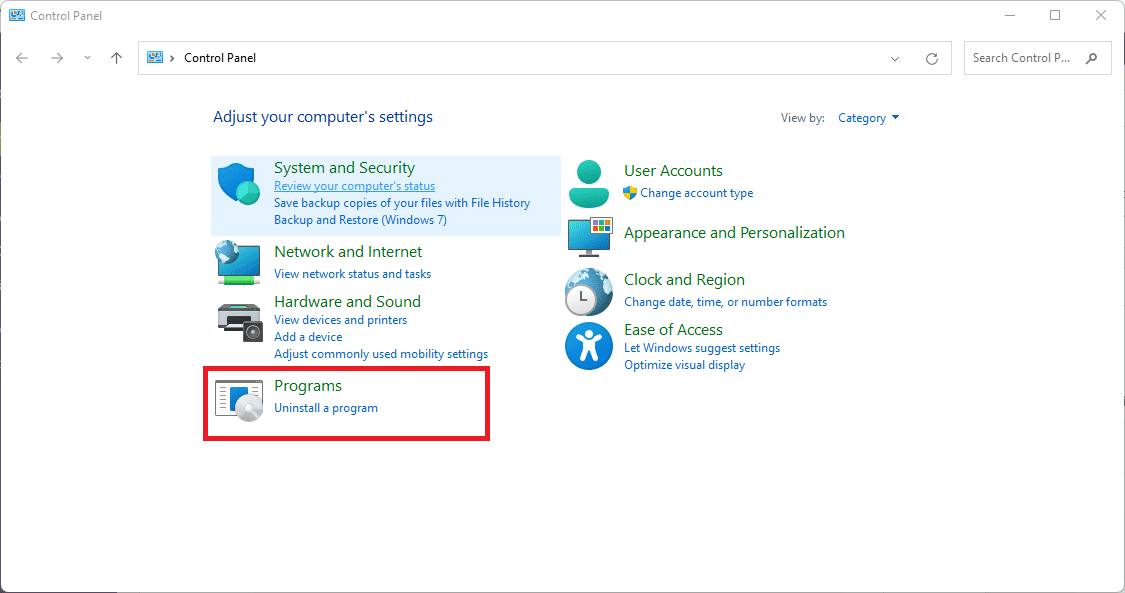
Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Windows-Subsystem-Linux



Alternatively, you can enable it via the “Control Panel” as illustrated below:

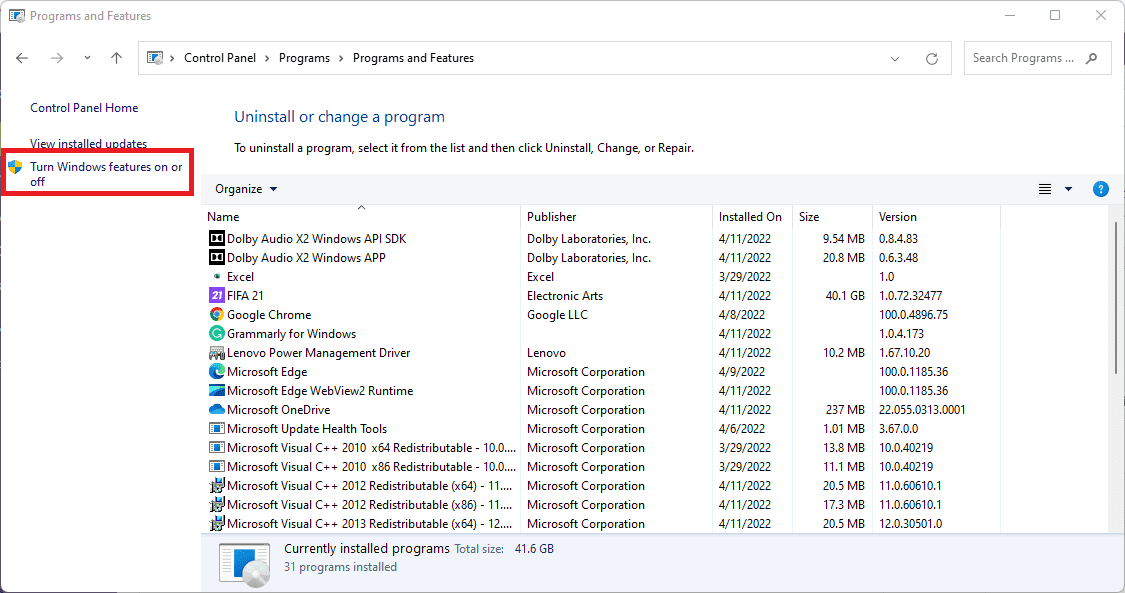
Open your system’s “Control Panel.”

Select “Programs” from the Control Panel.



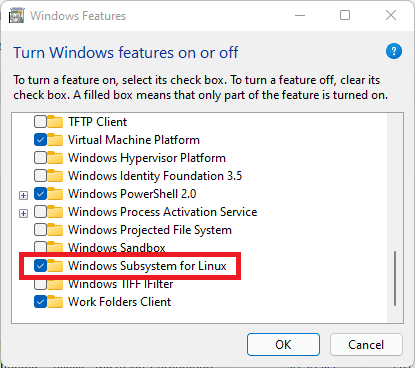
Programs

Click the “Turn Windows features on or off” option.



**Windows features**

Now, look for the “Windows Subsystem for Linux” function, tick it, and click “OK.” This operation will enable your Windows system’s WSL feature.



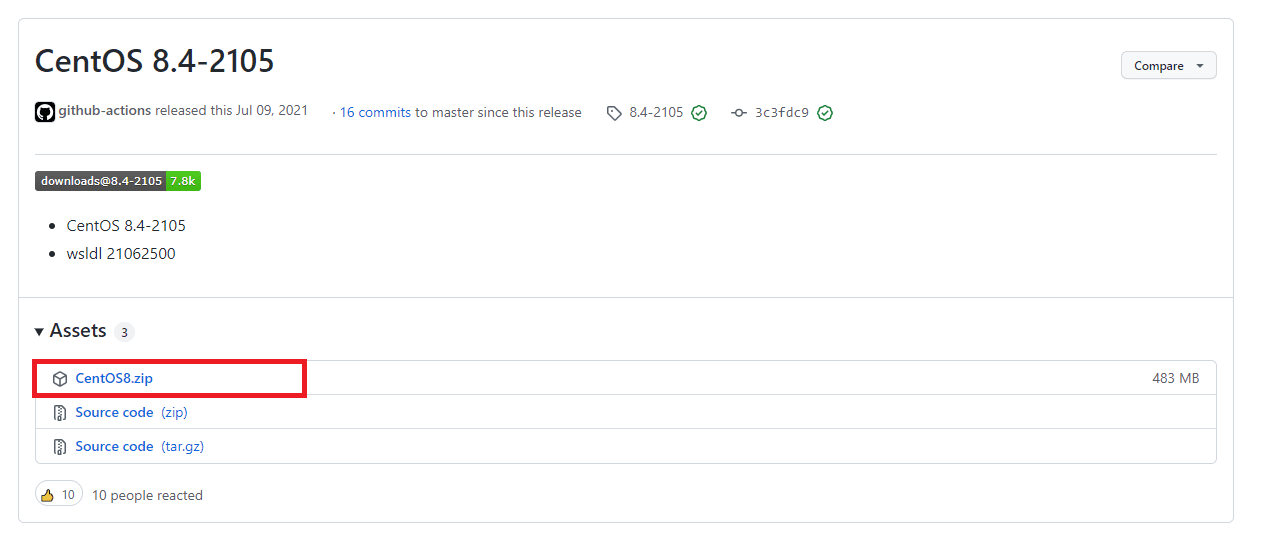
**Tick WSL**

Wait a few minutes while the WSL is installed on your system:

Now choose the “Restart now” button to apply the changes made and enable WSL on your Windows 10 PC.

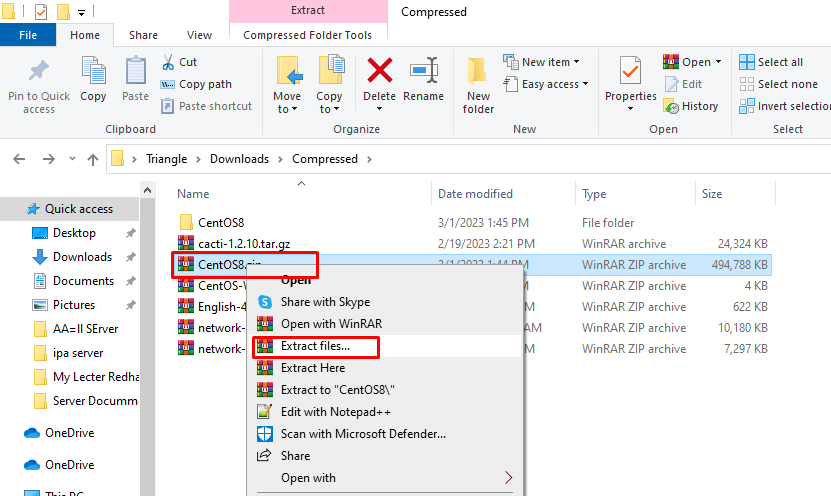
Step 2: Download CentOS WSL files

<https://github.com/mishamosher/CentOS-WSL/releases/tag/8.4-2105>



**Step 3: Extract CentOS WSL files**

Extract the “CentOS8” zip file on your PC by right-clicking on it and choosing the “Extract files” option

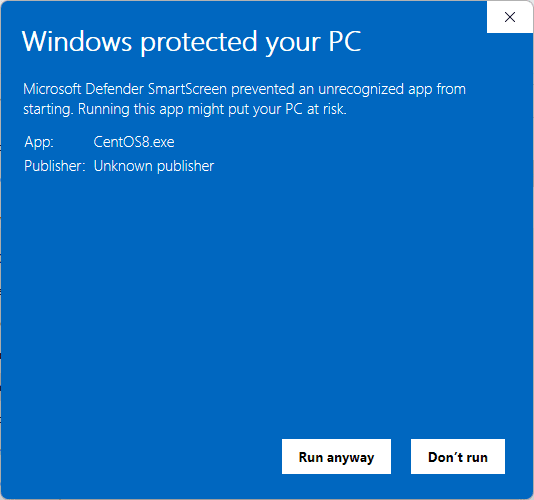


**Extract files**

Step 4: Install CentOS WSL

Start the CentOS8.exe file by opening the extracted folder. If Windows expresses doubt, pick the option “Run Anyway.”

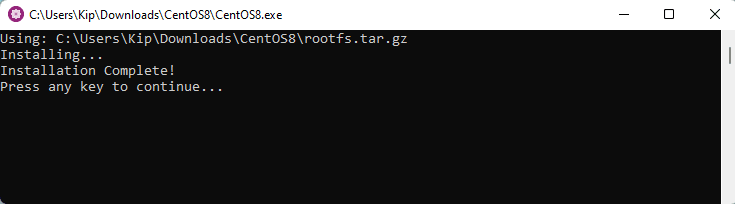


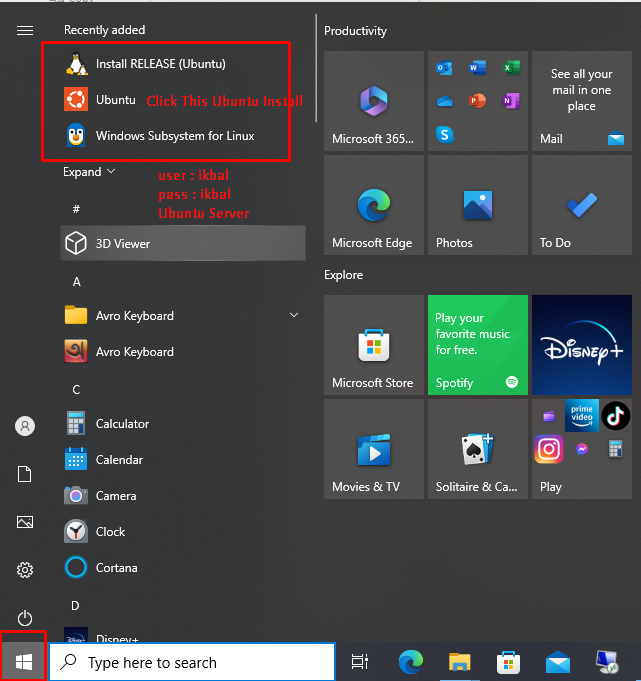


**Run anyway**

**Note:**Ensure you run the .exe file as an administrator.

Running the .exe file above will extract and register all essential files on your WSL.



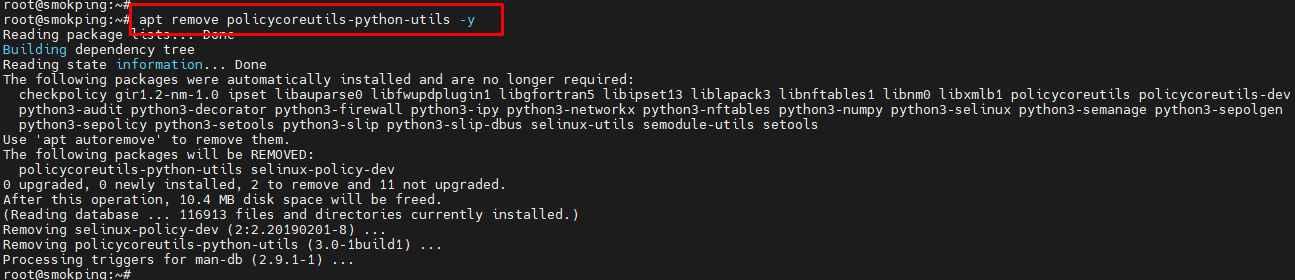


**Uninstalling CentOS8 from WSL**

./CentOS8.exe clean

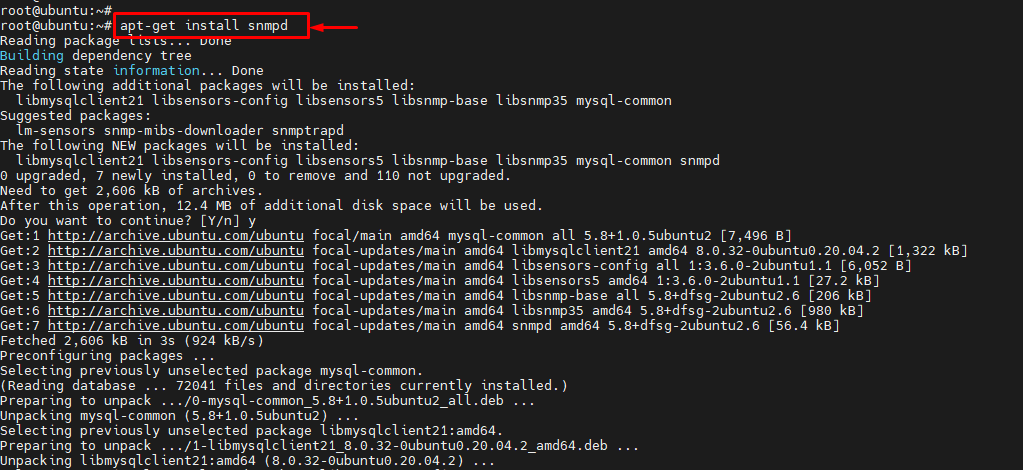
# [Uninstall Software Using the Command Line in Linux](https://www.howtogeek.com/229699/how-to-uninstall-software-using-the-command-line-in-linux/)

root@smokping:~# apt remove policycoreutils-python-utils –y



**Ubuntu: install and configure the SNMP service**

root@ubuntu:~# apt-get install snmpd



root@ubuntu:~# vim /etc/snmp/snmpd.conf

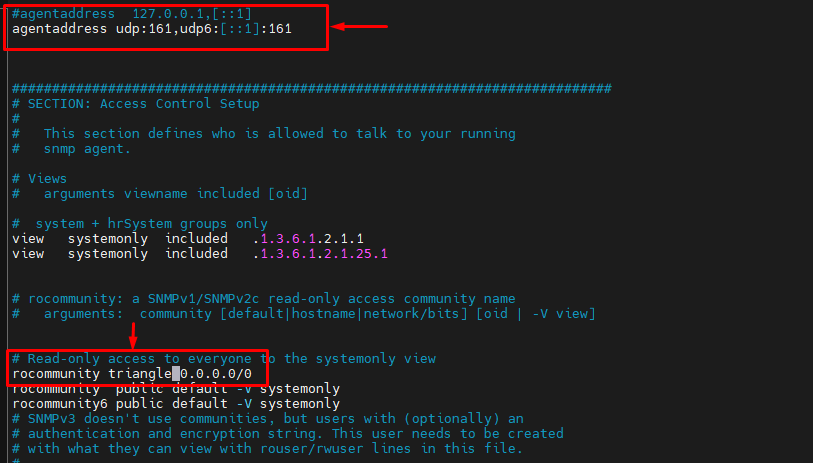
agentaddress udp:161,udp6:[::1]:161

rocommunity triangle 192.168.1.10/32

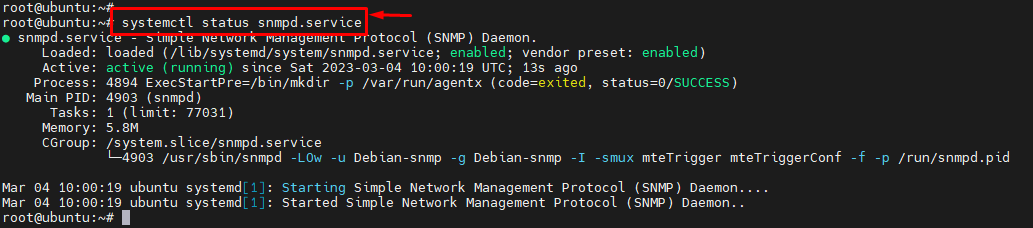
root@ubuntu:~# systemctl restart snmpd.service

root@ubuntu:~# systemctl start snmpd

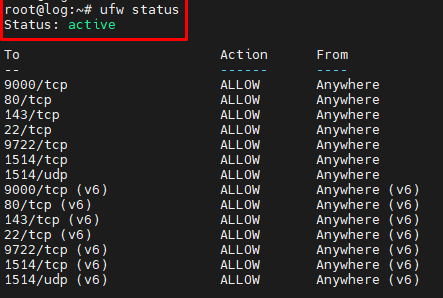
root@ubuntu:~# systemctl status snmpd.service



root@ubuntu:~# systemctl status snmpd.service



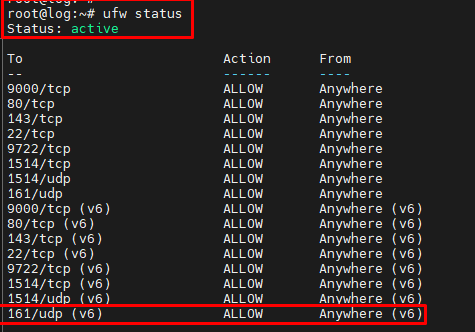
root@log:~# ufw status



root@log:~# ufw allow 161/udp



root@log:~# ufw status



root@log:~# ufw allow snmp



**SNMP Contact Information Add**

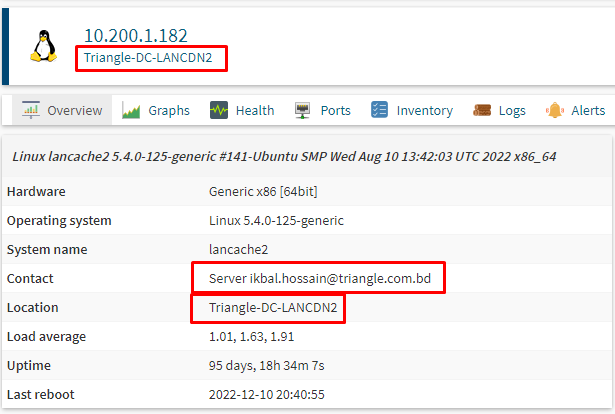
**####Ubuntu Server**

root@lancache2:~# cat /etc/snmp/snmpd.conf

rocommunity TrianglE ##SNMP Community Name Triangle\*/

syslocation Triangle-DC-LANCDN2 ##Server Location Set\*/

sysContact Server [ikbal.hossain@triangle.com.bd](mailto:ikbal.hossain@triangle.com.bd) ##Contact system Admin Info



**###Centos Server**

[root@graylog1 ~]# cat /etc/snmp/snmpd.conf

# sec.name source community

#com2sec notConfigUser default public

#rwcommunity triangle 0.0.0.0/0

# Change RANDOMSTRINGGOESHERE to your preferred SNMP community string

com2sec readonly default SNMP@dmin

com2sec readonly 0.0.0.0/0 triangle

group MyROGroup v2c readonly

group myGroup v2c myUser

view all included .1 80

access MyROGroup "" any noauth exact all none none

syslocation Elaach Content Server, Located in Datacenter, Supermicro, Rak no 2

syscontact Ikbal Hossain ikbal.hossain@triangle.com.bd

#Distro Detection

extend .1.3.6.1.4.1.2021.7890.1 distro /usr/bin/distro

pass .1.3.6.1.4.1.3582 /usr/sbin/lsi\_mrdsnmpmain

**Ubuntu network configuration**

root@ubuntu:~# vim /etc/netplan/00-installer-config.yaml

# This is the network config written by 'subiquity'

network:

ethernets:

ens1f0:

dhcp4: no

addresses: [162.254.198.254/30]

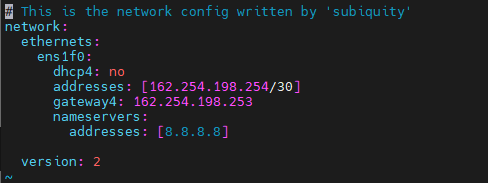
gateway4: 162.254.198.253

nameservers:

addresses: [8.8.8.8]

version: 2

~



root@ubuntu:~# netplan apply

root@ubuntu:~# netplan --debug apply

**Check Disk Performance (IOPS and Latency) in Linux?**

# yum install epel-release -y  
# yum install fio –y

Or apt-get in Debian or Ubuntu:

# apt-get install fio

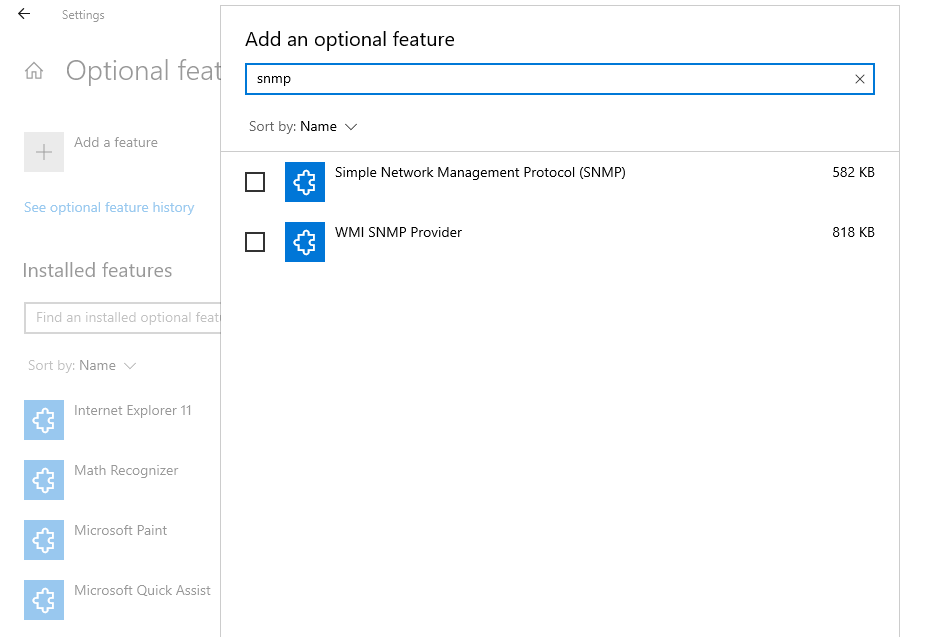
root@ubuntu:~# fio --randrepeat=1 --ioengine=libaio --direct=1 --gtod\_reduce=1 --name=fiotest --filename=testfio --bs=4k --iodepth=64 --size=8G --readwrite=randrw --rwmixread=75



**Install and Configure SNMP on Windows 10**

How to Enable SNMP From the Settings

SNMP is available as an optional feature on Windows 10. You can enable optional features by navigating to **Settings** > **Apps** > **Apps & features** > **Optional features**. Click on **Add a feature** and search for **snmp**. Select **Simple Network Management Protocol (SNMP)** and **WMI SNMP Provider** and click on **Install**.



# 10 Commands to Collect System and Hardware Info in Linux

### 1. How to View Linux System Information

**tecmint@tecmint ~ $** **uname**

To view your network hostname, use the **‘-n’** switch with the uname command as shown.

**tecmint@tecmint ~ $** **uname -n**

**tecmint.com**

To get information about kernel-version, use the **‘-v’** switch.

**tecmint@tecmint ~ $** **uname -v**

**#64-Ubuntu SMP Mon Sep 22 21:28:38 UTC 2014**

To get the information about your kernel release, use the **‘-r’** switch.

**tecmint@tecmint ~ $** **uname -r**

**3.13.0-37-generic**

To print your machine hardware name, use the **‘-m’** switch:

**tecmint@tecmint ~ $** **uname -m**

**x86\_64**

All this information can be printed at once by running the **‘uname -a’** command as shown below.

**tecmint@tecmint ~ $** **uname -a**

**Linux tecmint.com 3.13.0-37-generic #64-Ubuntu SMP Mon Sep 22 21:28:38**

**UTC 2014 x86\_64 x86\_64 x86\_64 GNU/Linux**

### 2. How to View Linux System Hardware Information

**tecmint@tecmint ~ $** **sudo lshw**

tecmint.com

description: Notebook

product: 20354 (LENOVO\_MT\_20354\_BU\_idea\_FM\_Lenovo Z50-70)

vendor: LENOVO

version: Lenovo Z50-70

serial: 1037407803441

width: 64 bits

capabilities: smbios-2.7 dmi-2.7 vsyscall32

configuration: administrator\_password=disabled boot=normal

chassis=notebook family=IDEAPAD frontpanel\_password=disabled

keyboard\_password=disabled power-on\_password=disabled

sku=LENOVO\_MT\_20354\_BU\_idea\_FM\_Lenovo Z50-70

uuid=E4B1D229-D237-E411-9F6E-28D244EBBD98

\*-core

description: Motherboard

product: Lancer 5A5

vendor: LENOVO

physical id: 0

version: 31900059WIN

serial: YB06377069

slot: Type2 - Board Chassis Location

\*-firmware

description: BIOS

vendor: LENOVO

physical id: 0

version: 9BCN26WW

date: 07/31/2014

size: 128KiB

capacity: 4032KiB

capabilities: pci upgrade shadowing cdboot bootselect edd

int13floppytoshiba int13floppy360 int13floppy1200 int13floppy720

int13floppy2880 int9keyboard int10video acpi usb biosbootspecification uefi

......

You can print a summary of your hardware information by using the **-short** option.

**tecmint@tecmint ~ $** **sudo lshw -short**

H/W path Device Class Description

=====================================================

system 20354 (LENOVO\_MT\_20354\_

BU\_idea\_FM\_Lenovo Z50-70)

/0 bus Lancer 5A5

/0/0 memory 128KiB BIOS

/0/4 processor Intel(R) Core(TM) i5-4210U

CPU @ 1.70GHz

/0/4/b memory 32KiB L1 cache

/0/4/c memory 256KiB L2 cache

/0/4/d memory 3MiB L3 cache

/0/a memory 32KiB L1 cache

/0/12 memory 8GiB System Memory

/0/12/0 memory DIMM [empty]

/0/12/1 memory DIMM [empty]

/0/12/2 memory 8GiB SODIMM DDR3 Synchronous

1600 MHz (0.6 ns)

/0/12/3 memory DIMM [empty]

/0/100 bridge Haswell-ULT DRAM Controller

/0/100/2 display Haswell-ULT Integrated

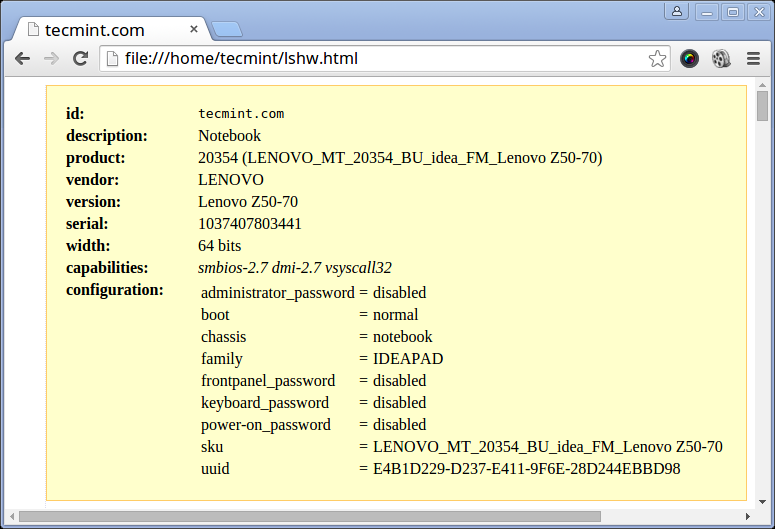
Graphics Controller

/0/100/3 multimedia Haswell-ULT HD Audio Controller

...

If you wish to generate output as an html file, you can use the option **-html**.

**tecmint@tecmint ~ $** **sudo lshw -html > lshw.html**

[](https://www.tecmint.com/wp-content/uploads/2015/09/Generate-Linux-Hardware-Information.png)Generate Linux Hardware Information in HTML

### 3. How to View Linux CPU Information

To view information about your CPU, use the [lscpu command](https://www.tecmint.com/check-linux-cpu-information/" \t "_blank" \o "Check Linux Architecture Info) as it shows information about your CPU architecture such as a number of CPUs, cores, CPU family model, CPU caches, threads, etc from **sysfs** and **/proc/cpuinfo**.

**tecmint@tecmint ~ $** **lscpu**

Architecture: x86\_64

CPU op-mode(s): 32-bit, 64-bit

Byte Order: Little Endian

CPU(s): 4

On-line CPU(s) list: 0-3

Thread(s) per core: 2

Core(s) per socket: 2

Socket(s): 1

NUMA node(s): 1

Vendor ID: GenuineIntel

CPU family: 6

Model: 69

Stepping: 1

CPU MHz: 768.000

BogoMIPS: 4788.72

Virtualization: VT-x

L1d cache: 32K

L1i cache: 32K

L2 cache: 256K

L3 cache: 3072K

NUMA node0 CPU(s): 0-3

### 4. How to Collect Linux Block Device Information

[Block devices](https://www.tecmint.com/create-disk-partitions-in-linux/) are storage devices such as hard disks, flash drives, etc. **lsblk** command is used to report information about block devices as follows.

**tecmint@tecmint ~ $** **lsblk**

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

sda 8:0 0 931.5G 0 disk

├─sda1 8:1 0 1000M 0 part

├─sda2 8:2 0 260M 0 part /boot/efi

├─sda3 8:3 0 1000M 0 part

├─sda4 8:4 0 128M 0 part

├─sda5 8:5 0 557.1G 0 part

├─sda6 8:6 0 25G 0 part

├─sda7 8:7 0 14.7G 0 part

├─sda8 8:8 0 1M 0 part

├─sda9 8:9 0 324.5G 0 part /

└─sda10 8:10 0 7.9G 0 part [SWAP]

sr0 11:0 1 1024M 0 rom

If you want to view all block devices on your system then include the **-a** option.

**tecmint@tecmint ~ $** **lsblk –a**

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

sda 8:0 0 931.5G 0 disk

├─sda1 8:1 0 1000M 0 part

├─sda2 8:2 0 260M 0 part /boot/efi

├─sda3 8:3 0 1000M 0 part

├─sda4 8:4 0 128M 0 part

├─sda5 8:5 0 557.1G 0 part

├─sda6 8:6 0 25G 0 part

├─sda7 8:7 0 14.7G 0 part

├─sda8 8:8 0 1M 0 part

├─sda9 8:9 0 324.5G 0 part /

└─sda10 8:10 0 7.9G 0 part [SWAP]

sdb 8:16 1 0 disk

sr0 11:0 1 1024M 0 rom

ram0 1:0 0 64M 0 disk

ram1 1:1 0 64M 0 disk

ram2 1:2 0 64M 0 disk

ram3 1:3 0 64M 0 disk

ram4 1:4 0 64M 0 disk

ram5 1:5 0 64M 0 disk

ram6 1:6 0 64M 0 disk

ram7 1:7 0 64M 0 disk

ram8 1:8 0 64M 0 disk

ram9 1:9 0 64M 0 disk

loop0 7:0 0 0 loop

loop1 7:1 0 0 loop

loop2 7:2 0 0 loop

loop3 7:3 0 0 loop

loop4 7:4 0 0 loop

loop5 7:5 0 0 loop

loop6 7:6 0 0 loop

loop7 7:7 0 0 loop

ram10 1:10 0 64M 0 disk

ram11 1:11 0 64M 0 disk

ram12 1:12 0 64M 0 disk

ram13 1:13 0 64M 0 disk

ram14 1:14 0 64M 0 disk

ram15 1:15 0 64M 0 disk

### 5. How to Print USB Controllers Information

The **lsusb** command is used to report information about USB controllers and all the devices that are connected to them.

**tecmint@tecmint ~ $** **lsusb**

Bus 001 Device 002: ID 8087:8000 Intel Corp.

Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub

Bus 003 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub

Bus 002 Device 005: ID 0bda:b728 Realtek Semiconductor Corp.

Bus 002 Device 004: ID 5986:0249 Acer, Inc

Bus 002 Device 003: ID 0bda:0129 Realtek Semiconductor Corp.

RTS5129 Card Reader Controller

Bus 002 Device 002: ID 045e:00cb Microsoft Corp.

Basic Optical Mouse v2.0

Bus 002 Device 001: ID 1d6b:0002 Linux Foundation

2.0 root hub

You can use the **-v** option to generate detailed information about each USB device.

**tecmint@tecmint ~ $** **lsusb -v**

### 6. How to Print PCI Devices Information

PCI devices may include usb ports, graphics cards, network adapters, etc. The **lspci tool** is used to generate information concerning all PCI controllers on your system plus the devices that are connected to them.

To print information about PCI devices run the following command.

**tecmint@tecmint ~ $** **lspci**

00:00.0 Host bridge: Intel Corporation Haswell-ULT

DRAM Controller (rev 0b)

00:02.0 VGA compatible controller: Intel Corporation Haswell-ULT

Integrated Graphics Controller (rev 0b)

00:03.0 Audio device: Intel Corporation Haswell-ULT HD Audio Controller

(rev 0b)

00:14.0 USB controller: Intel Corporation Lynx Point-LP USB xHCI HC

(rev 04)

00:16.0 Communication controller: Intel Corporation Lynx Point-LP HECI #0

(rev 04)

00:1b.0 Audio device: Intel Corporation Lynx Point-LP HD Audio Controller

(rev 04)

00:1c.0 PCI bridge: Intel Corporation Lynx Point-LP PCI Express Root Port 3

(rev e4)

00:1c.3 PCI bridge: Intel Corporation Lynx Point-LP PCI Express Root Port 4

(rev e4)

00:1c.4 PCI bridge: Intel Corporation Lynx Point-LP PCI Express Root Port 5

(rev e4)

00:1d.0 USB controller: Intel Corporation Lynx Point-LP USB EHCI #1

(rev 04)

00:1f.0 ISA bridge: Intel Corporation Lynx Point-LP LPC Controller

(rev 04)

00:1f.2 SATA controller: Intel Corporation Lynx Point-LP SATA Controller 1

[AHCI mode] (rev 04)

00:1f.3 SMBus: Intel Corporation Lynx Point-LP SMBus Controller (rev 04)

01:00.0 Ethernet controller: Realtek Semiconductor Co., Ltd. RTL8111/8168/8411

PCI Express Gigabit Ethernet Controller (rev 10)

02:00.0 Network controller: Realtek Semiconductor Co., Ltd.

RTL8723BE PCIe Wireless Network Adapter

03:00.0 3D controller: NVIDIA Corporation GM108M [GeForce 840M] (rev a2)

Use the **-t** option to produce output in a tree format.

**tecmint@tecmint ~ $** **lspci -t**

-[0000:00]-+-00.0

+-02.0

+-03.0

+-14.0

+-16.0

+-1b.0

+-1c.0-[01]----00.0

+-1c.3-[02]----00.0

+-1c.4-[03]----00.0

+-1d.0

+-1f.0

+-1f.2

\-1f.3

Use the **-v** option to produce detailed information about each connected device.

**tecmint@tecmint ~ $** **lspci -v**

00:00.0 Host bridge: Intel Corporation Haswell-ULT DRAM Controller (rev 0b)

Subsystem: Lenovo Device 3978

Flags: bus master, fast devsel, latency 0

Capabilities:

00:02.0 VGA compatible controller: Intel Corporation Haswell-ULT

Integrated Graphics Controller (rev 0b) (prog-if 00 [VGA controller])

Subsystem: Lenovo Device 380d

Flags: bus master, fast devsel, latency 0, IRQ 62

Memory at c3000000 (64-bit, non-prefetchable) [size=4M]

Memory at d0000000 (64-bit, prefetchable) [size=256M]

I/O ports at 6000 [size=64]

Expansion ROM at [disabled]

Capabilities:

Kernel driver in use: i915

.....

### 7. How to Print SCSI Devices Information

To view all your scsi/sata devices, use the **lsscsi** command as follows. If you do not have the **lsscsi** tool installed, run the following command to install it.

$ sudo apt-get install lsscsi [on **Debian** derivatives]

# yum install lsscsi [On RedHat based systems]

# dnf install lsscsi [On Fedora 21+ Onwards]

After installation, run the **lsscsi** command as shown:

**tecmint@tecmint ~ $** **lsscsi**

[0:0:0:0] disk ATA ST1000LM024 HN-M 2BA3 /dev/sda

[1:0:0:0] cd/dvd PLDS DVD-RW DA8A5SH RL61 /dev/sr0

[4:0:0:0] disk Generic- xD/SD/M.S. 1.00 /dev/sdb

Use the **-s** option to show device sizes.

**tecmint@tecmint ~ $** **lsscsi -s**

[0:0:0:0] disk ATA ST1000LM024 HN-M 2BA3 /dev/sda 1.00TB

[1:0:0:0] cd/dvd PLDS DVD-RW DA8A5SH RL61 /dev/sr0 -

[4:0:0:0] disk Generic- xD/SD/M.S. 1.00 /dev/sdb -

### 8. How to Print Information about SATA Devices

You can find some information about sata devices on your system as follows using the **hdparm** utility. In the example below, I used the block device **/dev/sda1** which is the hard disk on my system.

**tecmint@tecmint ~ $** **sudo hdparm /dev/sda1**

[root@localhost ~]# **sudo hdparm /dev/nvme0n1**

/dev/sda1:

multcount = 0 (off)

IO\_support = 1 (32-bit)

readonly = 0 (off)

readahead = 256 (on)

geometry = 56065/255/63, sectors = 2048000, start = 2048

To print information about device geometry in terms of cylinders, heads, sectors, size, and the starting offset of the device, use the **-g** option.

**tecmint@tecmint ~ $** **sudo hdparm -g /dev/sda1**

/dev/sda1:

geometry = 56065/255/63, sectors = 2048000, start = 2048

### 9. How to Check Linux File System Information

To gather information about file system partitions, you can use the [fdisk command](https://www.tecmint.com/fdisk-commands-to-manage-linux-disk-partitions/" \t "_blank" \o "Linux disk-partitioning). Although the main functionality of the **fdisk** command is to [modify file system partitions](https://www.tecmint.com/linux-tools-to-monitor-disk-partition-usage/), it can also be used to view information about the different partitions on your file system.

You can print partition information as follows. Remember to run the command as a superuser or else you may not see any output.

**tecmint@tecmint ~ $** **sudo fdisk -l**

WARNING: GPT (GUID Partition Table) detected on '/dev/sda'!

The util fdisk doesn't support GPT. Use GNU Parted.

Disk /dev/sda: 1000.2 GB, 1000204886016 bytes

255 heads, 63 sectors/track, 121601 cylinders,

total 1953525168 sectors

Units = sectors of 1 \* 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 4096 bytes

I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk identifier: 0xcee8ad92

Device Boot Start End Blocks Id System

/dev/sda1 1 1953525167 976762583+ ee GPT

Partition 1 does not start on physical sector boundary.

### 10. How to Check Linux Hardware Components Info

You can also use the [dmidecode utility](https://www.tecmint.com/how-to-get-hardware-information-with-dmidecode-command-on-linux/" \t "_blank" \o "Get Linux Hardware Info) to extract hardware information by reading data from the DMI tables.

To print information about memory, run this command as a superuser.

**tecmint@tecmint ~ $** **sudo dmidecode -t memory**

# dmidecode 2.12

# SMBIOS entry point at 0xaaebef98

SMBIOS 2.7 present.

Handle 0x0005, DMI type 5, 24 bytes

Memory Controller Information

Error Detecting Method: None

Error Correcting Capabilities:

None

Supported Interleave: One-way Interleave

Current Interleave: One-way Interleave

Maximum Memory Module Size: 8192 MB

Maximum Total Memory Size: 32768 MB

Supported Speeds:

Other

Supported Memory Types:

Other

Memory Module Voltage: Unknown

Associated Memory Slots: 4

0x0006

0x0007

0x0008

0x0009

Enabled Error Correcting Capabilities:

None

To print information about the system, run this command.

**tecmint@tecmint ~ $** **sudo dmidecode -t system**

# dmidecode 2.12

# SMBIOS entry point at 0xaaebef98

SMBIOS 2.7 present.

Handle 0x0001, DMI type 1, 27 bytes

System Information

Manufacturer: LENOVO

Product Name: 20354

Version: Lenovo Z50-70

Serial Number: 1037407803441

UUID: 29D2B1E4-37D2-11E4-9F6E-28D244EBBD98

Wake-up Type: Power Switch

SKU Number: LENOVO\_MT\_20354\_BU\_idea\_FM\_Lenovo Z50-70

Family: IDEAPAD

...

To print information about BIOS, run this command.

**tecmint@tecmint ~ $** **sudo dmidecode -t bios**

# dmidecode 2.12

# SMBIOS entry point at 0xaaebef98

SMBIOS 2.7 present.

Handle 0x0000, DMI type 0, 24 bytes

BIOS Information

Vendor: LENOVO

Version: 9BCN26WW

Release Date: 07/31/2014

Address: 0xE0000

Runtime Size: 128 kB

ROM Size: 4096 kB

Characteristics:

PCI is supported

BIOS is upgradeable

BIOS shadowing is allowed

Boot from CD is supported

Selectable boot is supported

EDD is supported

Japanese floppy for NEC 9800 1.2 MB is supported (int 13h)

Japanese floppy for Toshiba 1.2 MB is supported (int 13h)

5.25"/360 kB floppy services are supported (int 13h)

5.25"/1.2 MB floppy services are supported (int 13h)

3.5"/720 kB floppy services are supported (int 13h)

3.5"/2.88 MB floppy services are supported (int 13h)

8042 keyboard services are supported (int 9h)

CGA/mono video services are supported (int 10h)

ACPI is supported

USB legacy is supported

BIOS boot specification is supported

Targeted content distribution is supported

UEFI is supported

BIOS Revision: 0.26

Firmware Revision: 0.26

...

To print information about the processor, run this command.

**tecmint@tecmint ~ $** **sudo dmidecode -t processor**

# dmidecode 2.12

# SMBIOS entry point at 0xaaebef98

SMBIOS 2.7 present.

Handle 0x0004, DMI type 4, 42 bytes

Processor Information

Socket Designation: U3E1

Type: Central Processor

Family: Core i5

Manufacturer: Intel(R) Corporation

ID: 51 06 04 00 FF FB EB BF

Signature: Type 0, Family 6, Model 69, Stepping 1

Flags:

...

**BIOS Keys by Manufacturer.**



## BIOS Keys by Manufacturer

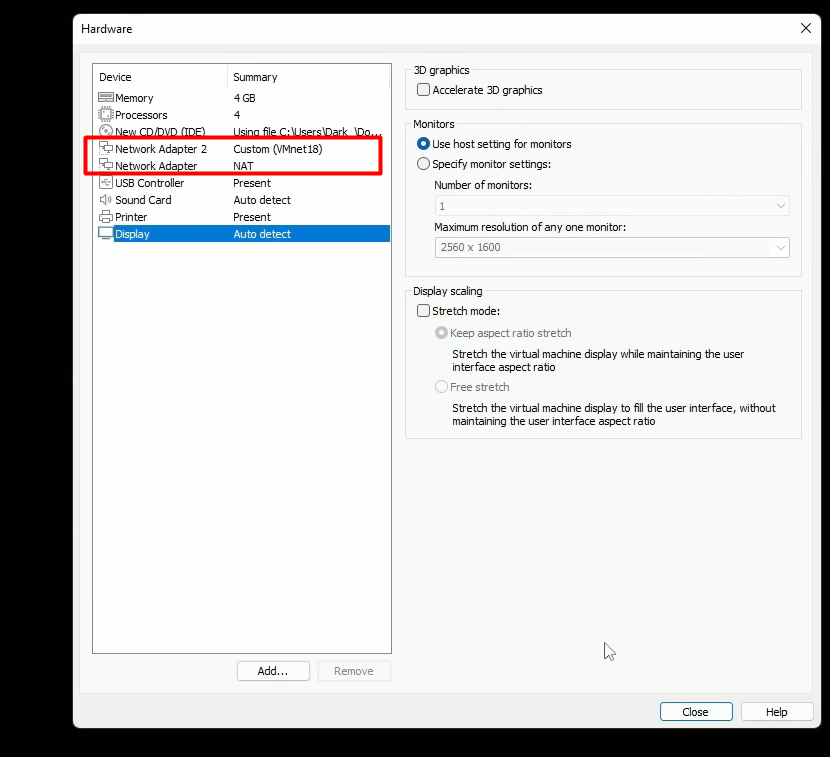
Here's a list of common BIOS keys by brand. Depending on the age of your model, the key may be different.

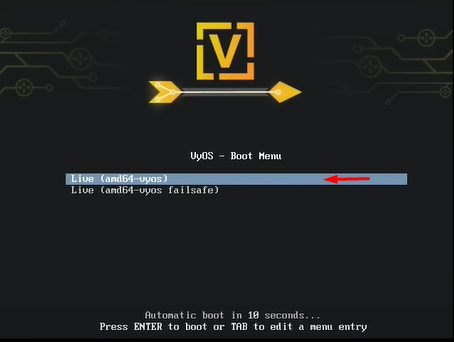
* **ASRock:**F2 or DEL
* **ASUS:**F2 for all PCs, F2 or DEL for Motherboards
* **Acer:**F2 or DEL
* **Dell:**F2 or F12
* **ECS:**DEL
* **Gigabyte / Aorus:** F2 or DEL
* **HP:** F10
* **Lenovo (Consumer Laptops):**F2 or Fn + F2
* **Lenovo (Desktops):**F1
* **Lenovo (ThinkPads):**Enter then F1.
* **MSI:**DEL for motherboards and PCs
* **Microsoft Surface Tablets:**Press and hold volume up button.
* **Origin PC:** F2
* **Samsung:**F2
* **Toshiba:**F2
* **Zotac:** DEL

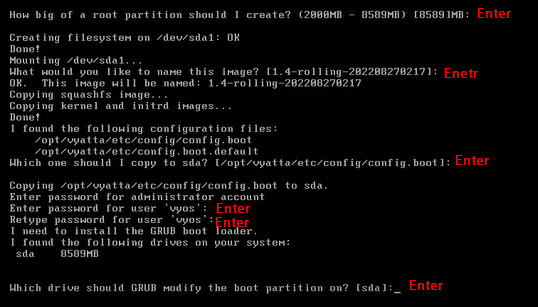
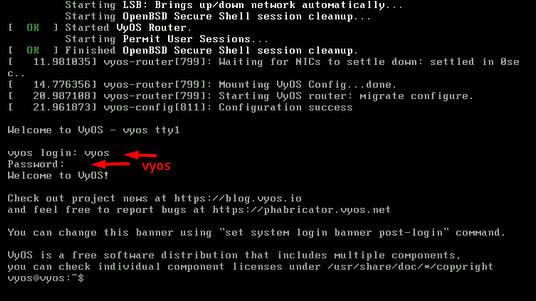
Format a Hard Drive Using the Command Prompt

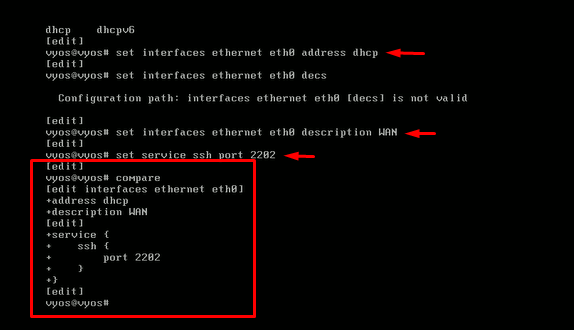
**Vyos Router install and Configuration Command**











ssh vyos@192.168.30.147 -p 9722

vyos@vyos# set system host-name router

vyos@router# set interfaces ethernet eth1 address 172.16.5.1/24

**DHCP Server**

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 lease 86400

[edit]

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 name-server 8.8.8.8

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 domain-name tnb.local

[edit]

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 default-router 172.16.5.1

[edit]

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 range 0 start 172.16.5.50

[edit]

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 range 0 stop 172.16.5.100

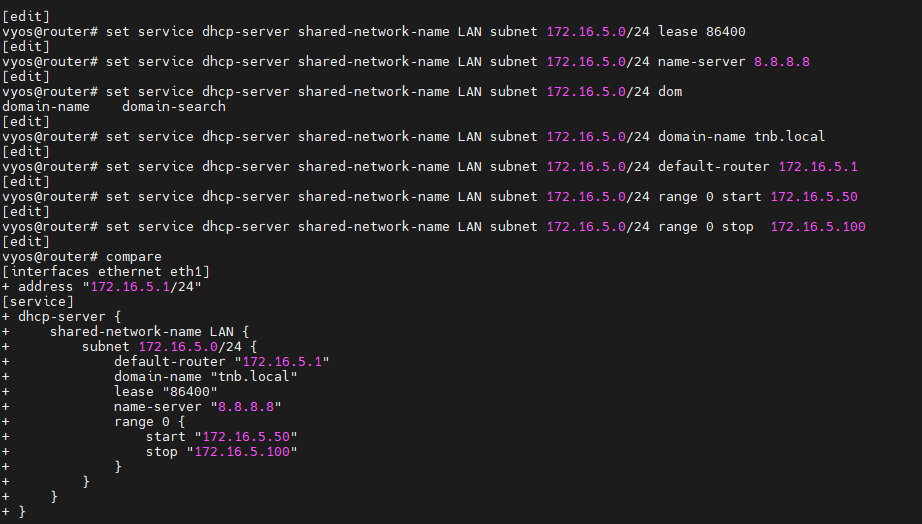
[edit]

vyos@router# compare

vyos@router# commit

[edit]

vyos@router# save



NAT RULES

vyos@router# set nat source rule 100 source address 172.16.5.0/24

vyos@router# set nat source rule 100 outbound-interface eth0

vyos@router# set nat source rule 100 translation address masquerade

vyos@router# commit

[edit]

vyos@router# save

**Vyos Router Command Line**

vyos@vyos $ show interface

vyos@vyos $ config

===================================== router Host Name Change ==================

vyos@vyos # set system host-name router

=======================Set Router Interface Ip Add Static and DHCP ==================

vyos@router # set int ethernet eth0 address dhcp

vyos@router # set int ethernet eth0 address 10.200.6.22/28

vyos@router # set int ethernet eth0 description Internet

vyos@router # set int ethernet eth1 address 172.16.5.1/24

vyos@router # set int ehternet eth1 description LAN

============================Router Interface Ip Delete Command ==================

vyos@router # delete int ethernet eth0 address 10.200.6.22/28

vyos@router # delete int ethernet eth0 description Internet

===================================== Set Static Route ==================

vyos@router # set protocols static route 0.0.0.0/0 next-hop 10.200.6.17 //Gateway ip address

===================================== Delete Static Route ==================

vyos@router # delete protocols static route 0.0.0.0/0

===================================== Show Route ==================

tsl@tsl-vyos-r1:~$ show ip route

================================DHCP Server ==============================

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 lease 86400

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 name-server 8.8.8.8

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 domain-name tnb.local

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 default-router 172.16.5.1

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 range 0 start 172.16.5.50

vyos@router# set service dhcp-server shared-network-name LAN subnet 172.16.5.0/24 range 0 stop 172.16.5.100

vyos@router# compare

vyos@router# commit

vyos@router# save

=====================================Nat Rules=======================

vyos@router# set nat source rule 100 source address 172.16.5.0/24

vyos@router# set nat source rule 100 outbound-interface eth0

vyos@router# set nat source rule 100 translation address masquerade

vyos@router# compare

vyos@router# commit

vyos@router# save

==================================Nat Rules Delete =======================

tsl@tsl-vyos-r1# delete nat source rule 100

=======================================Show Command ==============

IP configuration ==========> show interfaces

Routing ===================> show ip route

Show configuration ========> show

tsl@tsl-vyos-r1:~$ show log tail

tsl@tsl-vyos-r1:~$ monitor log ssh

tsl@tsl-vyos-r1:~$ monitor traffic interface eth0

ikbal@router:~$ monitor bandwidth interface eth1

vyos@router $ configure

vyos@vyos # [configuration commands] Global Mode

vyos@vyos # commit

vyos@vyos # save

vyos@vyos # exit

vyos@vyos $

User: tsl

pass: tsl123

====================== New User Create and login Password ========================>

vyos@vyos:~$ config

vyos@router # set system login user [username] authentication plaintext-password [password]

vyos@router# set system login user ikbal authentication plaintext-password ikbal

vyos@router # save

vyos@router # exit

vyos@vyos-rtr:~$ configure

yos@router# set system login user ikbal full-name "Ikbal Hossain"

vyos@vyos-rtr# set system login user ikbal authentication plaintext-password ikbal

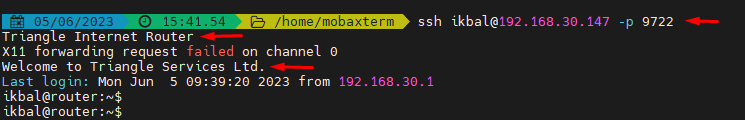
vyos@vyos-rtr# set system login user ikbal level admin

vyos@vyos-rtr# commit

vyos@vyos-rtr# save

ikbal@router# set system login banner post-login "Welcome to Triangle Services Ltd."

vyos@router# set system login banner pre-login "Triangle Internet Router"



================================= Set SSH Command ========================>

vyos@vyos-rtr# set service ssh port 9722

================================= Router off command ========================>

vyos@vyos:~$ poweroff

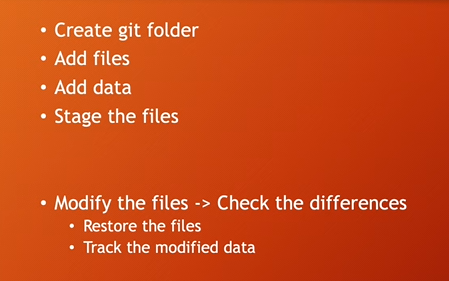
vyos@vyos:~$ reboot

**VMware Tools Install**

# yum install open-vm-tools -y

use this command to install vmware tools

**Git & Github**



**User name create & Password create:**

$ git config --global user.name “[ikbalbd1052@gmail.com](mailto:ikbalbd1052@gmail.com)”

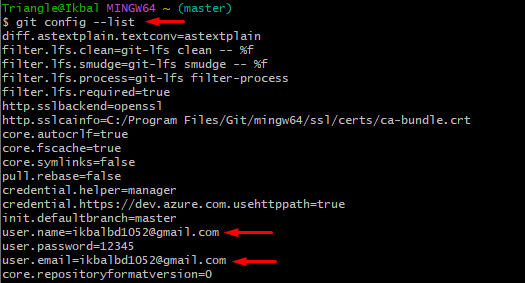
$ git config --global user.email [ikbalbd1052@gmail.com](mailto:ikbalbd1052@gmail.com)

**Git User Name Modify:**

$ git config --global user.name "ikbal"

**Git all config show:**

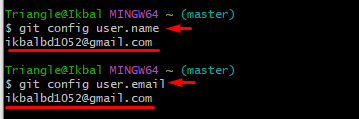
$ git config –list



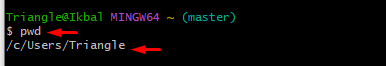
**Show Git User Name & User mail:**

$ git config user.name

$ git config user.email



$ pwd



$ cd Desktop/

$ pwd

$ mkdir nots

$ cd nots/

$ ls -a

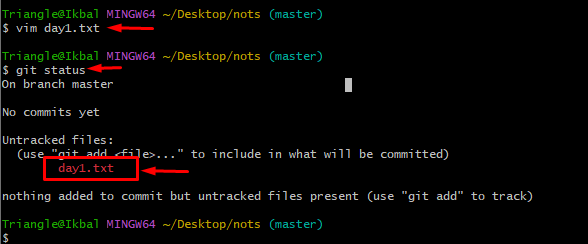
$ git init



$ ls -a

$ touch day1.txt

$ vim day1.txt

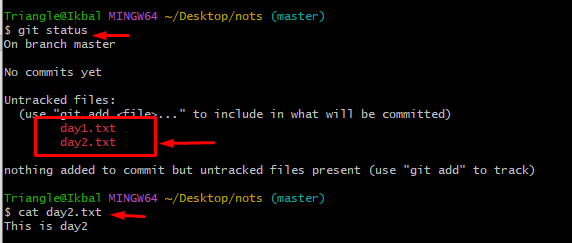


$ touch day2.txt

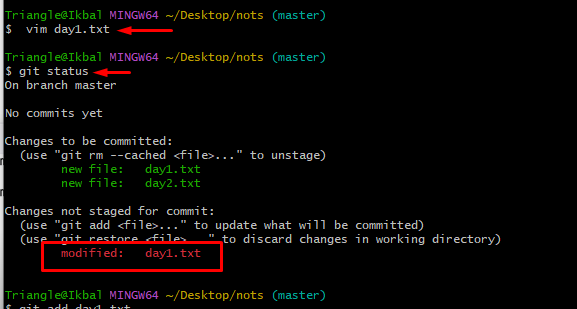
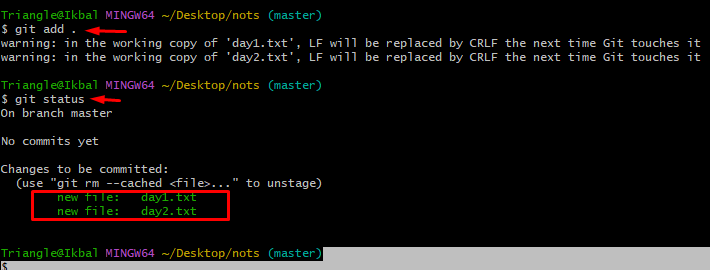
$ ls

$ vim day2.txt

$ git status



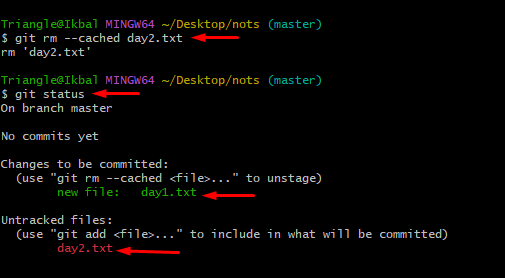
$ git add .



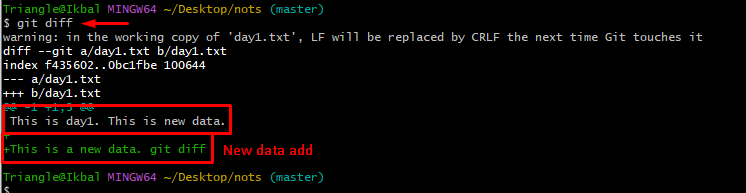
$ git add day1.txt

$ git restore day2.txt

$ git rm --cached day2.txt



$ git diff



======================Git Command==========================

* git init -- creates the local repo and initialized
* git clone link -- creates the clone folder
* git status -- shows modified files
* git add -all - add the directory change to stage
* git add . -- add the specific directory change to stage
* git add \* -- add all files change except delete
* git add link -- add specific change file
* git reset -- resets the changes of git add ]
* git commit m "message" - commit the changes
* git reset HEAD~ -- reset all commits
* git branch name -- creates a git branch of the following name
* git checkout name -- goes to the name branch
* git branch -- lists all the branch
* git merge name -- transfer properities of name to directory
* git push orgin name -- transfer the folder to the online github repo
* git pull -- git fetch plus merge

IP Core Networks

====================================================

Hope this may help. By the way, Thanks dada!

Commands:

* 1) touch filename.extension --to create a file
* 2) mkdir foldername -- to create a folder
* 2.1) cd foldername -- to enter a folder
* 3) git init -- to create a git in your working directory
* 4) git clone URL -- to pull a repository from the cloud
* 5) git status -- to check the changes in your file or to check the current status
* 6) pwd -- to check the directory currently you are working
* 7) ”git add -all” or “git add -A” - to stage your root folder
* 7.1) ” git add . ” -- to stage changes in your folder you are currently staying
* 7.2) git add \* -- to stage changes all the files except the deleted one
* 7.3) git add \*.extension -- to stage changes of all the files of the specific extension
* 8) git reset -- to unstage your changes in your files
* 9) git commit -m”write something here” --to commit your staged changes
* 10) git reset HEAD~ -- to unstage your committed changes
* 11) git reset -hard - almost same to the “git reset” but it also gives you the deleted files
* 12) git rm filename.extension -- to delete and stage the changes in your file
* 12.1) git rm filename.extension f - to delete the file forcefully which hasn’t been staged
* 12.2) git rm --cached filename.extension --to stage the changes and not to delete the file from working directory
* 12.2)git rm -r folder ( google it)
* Branch
* 13) git branch ---to check the current branch
* 14)git branch branch-name -- to create a new branch
* 15)git checkout branchName -- to switch to a branch
* 16) git merge branchName -m”write something” --to merge your branch with another branch
* 17) ls --to see the list of your files
* Git push, fetch, pull
* 18) git push origin branchName
* 19) git fetch
* 20) git pull

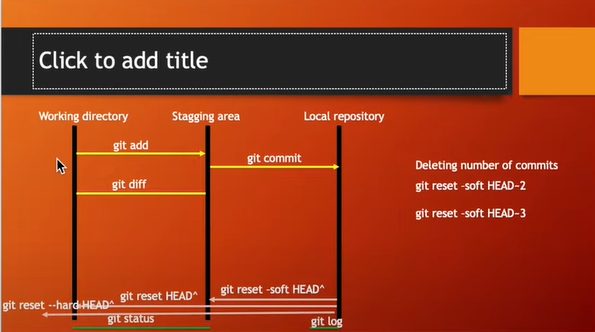
**#### Git Commit && Reset:**

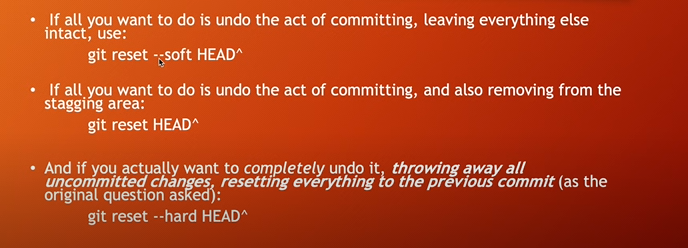


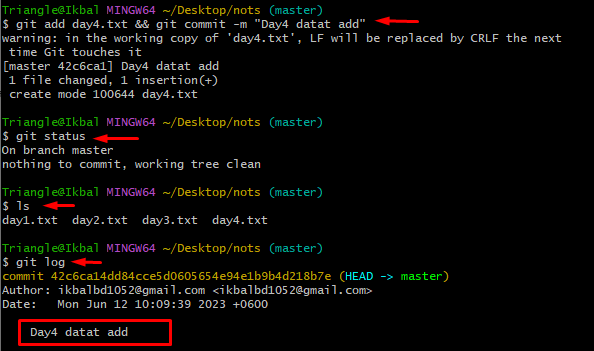
$ git commit -m 'day1 data is added'

$ git log

$ git add day4.txt && git commit -m "Day4 datat add"

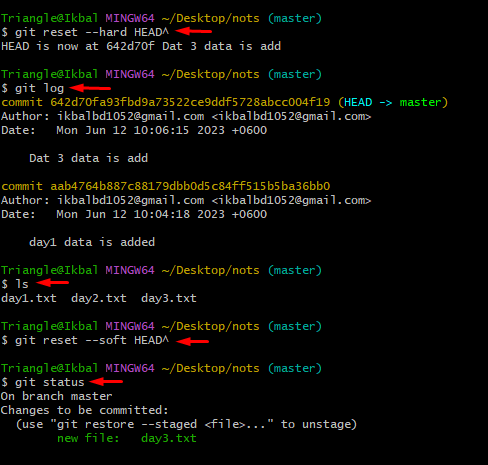






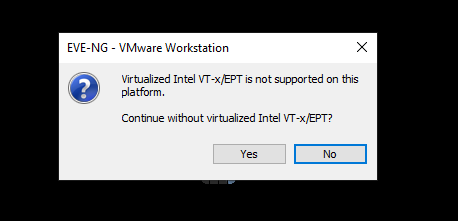
$ git reset --hard HEAD^

$ git reset --soft HEAD^



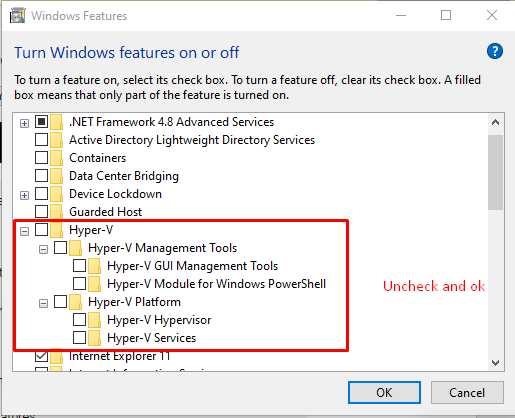
$ git reset HEAD^

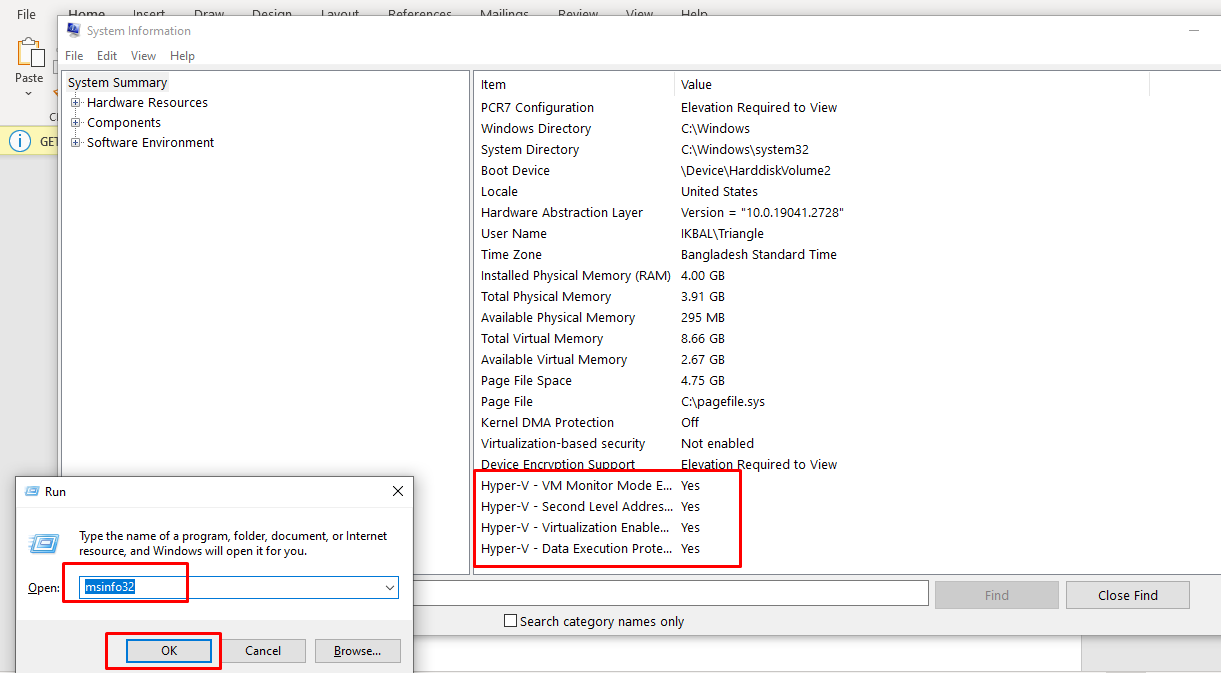
**virtualized intel vt-x/ept is not supported: Solution**





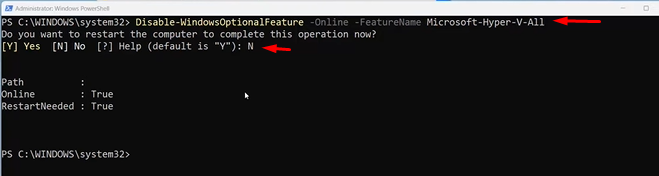
Check Your Hyper-V enabled.







1st Command: bcdedit /set hypervisorlaunchtype off”



2nd Command: Disable-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V-All

Source link : <https://www.youtube.com/watch?v=6f1Qckg2Zx0&ab_channel=ProgrammingKnowledge2>

**Linux Server Core Check Command**

root@tsl-vyos:~# cat /proc/cpuinfo | grep vmx | wc -l

48

root@tsl-vyos:~# grep processor /proc/cpuinfo | wc -l

24

root@tsl-vyos:~# dmidecode | grep -i HTT

HTT (Multi-threading)

HTT (Multi-threading)

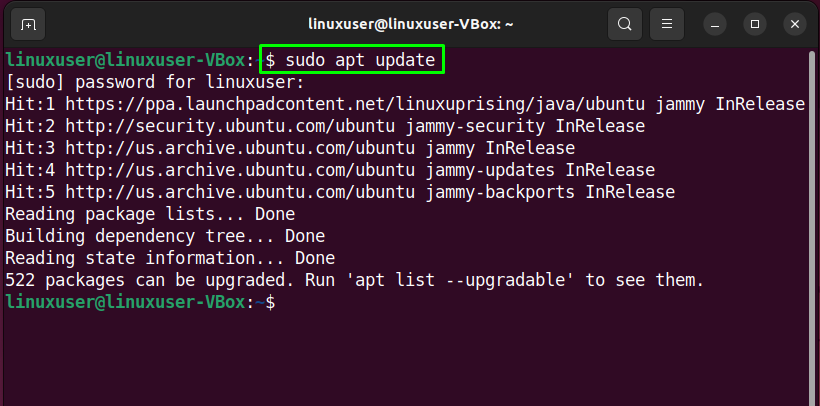
Invalid entry length (16). Fixed up to 11.

root@tsl-vyos:~# cat /proc/cpuinfo | grep -o ht | uniq

ht

**Advanced UFW Firewall Configuration in Ubuntu 22.04**

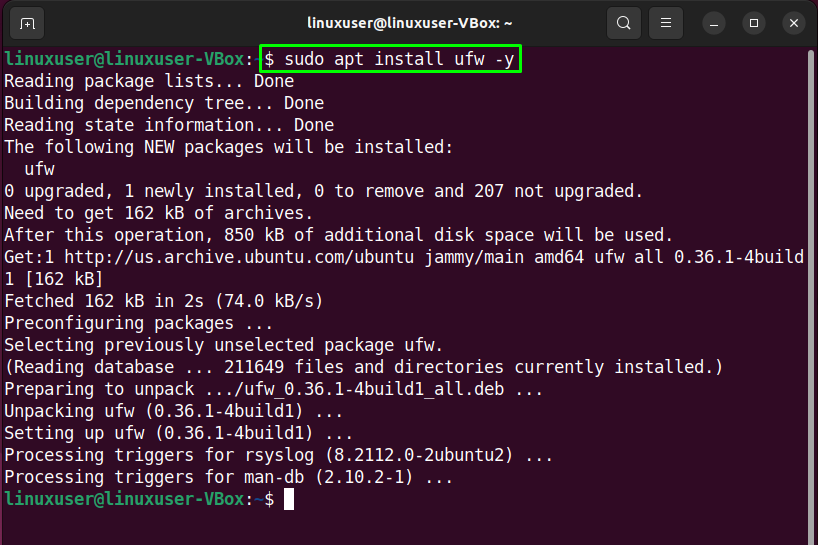
*$*sudo apt update



### ****Step 2: Install UFW****

Next, execute the provided command for the installation of UFW:

*$*sudo apt install ufw -y

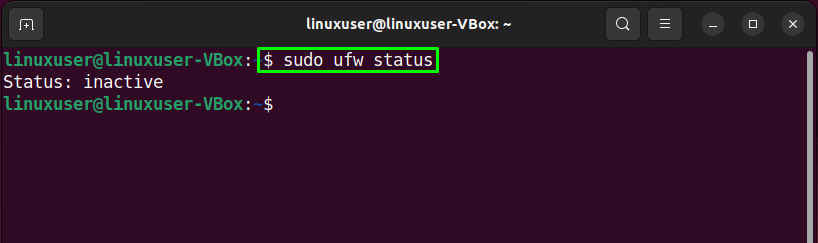


### ****Step 3: Check UFW status****

After successfully installing UFW, check out its status on your system:

*$*sudo ufw status

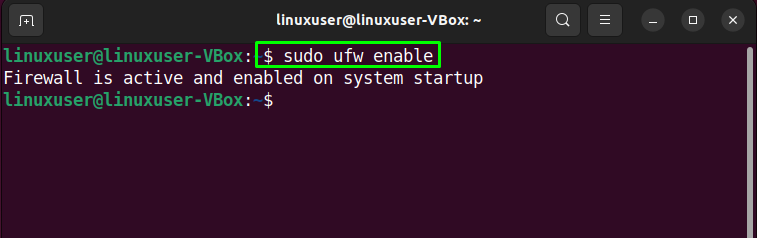
The given output indicates that UFW is currently “**inactive**” on our Ubuntu 22.04:



### ****Step 4: Enable UFW****

Now, it’s time to enable UFW with the help of the following command:

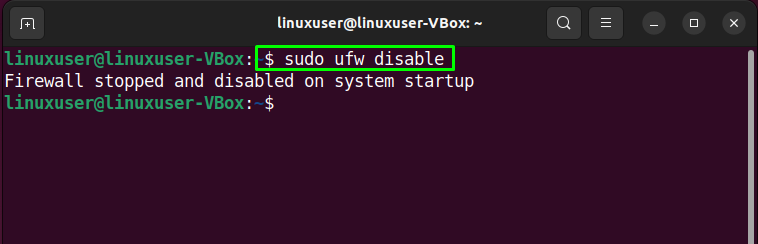
*$*sudo ufw enable



### ****Step 5: Disable UFW****

In case, if you want to disable UFW, then utilize the below-given command in your Ubuntu 22.04 terminal:

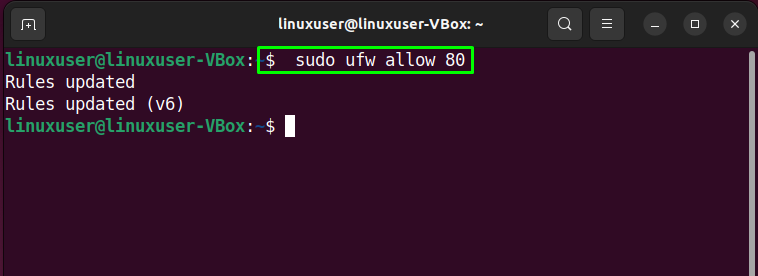
*$*sudo ufw disable



### ****Step 6: Allow traffic****

Ubuntu 22.04 permits you to define UFW rules using service name or port number. For instance, to **allow incoming connections**for port “**80**”, we will execute the following command:

*$*sudo ufw allow 80

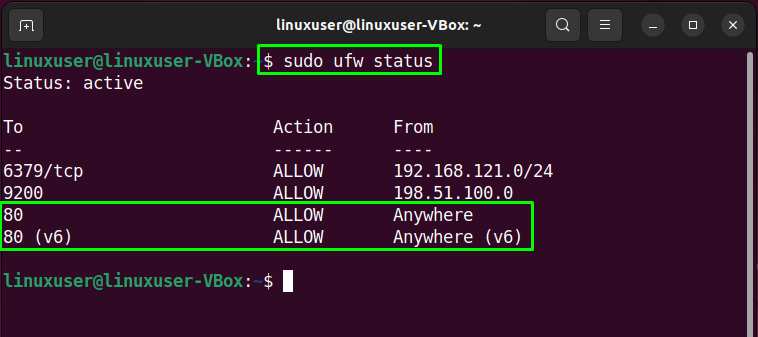


### ****Step 7: Check status****

Again, verifying UFW status will let you know if the proceeding operation was successfully performed:

*$*sudo ufw status

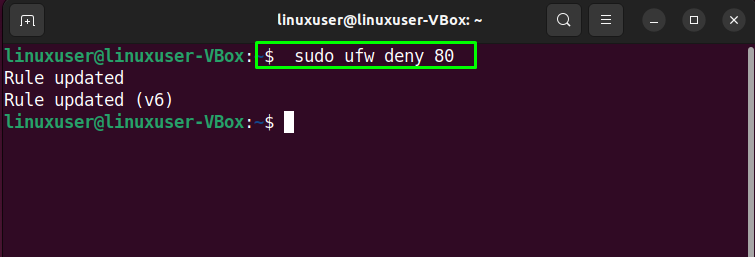
As you can see, port “**80**” is allowed to accept incoming connections from “**Anywhere**”:



### ****Step 8: Deny traffic****

Similarly, the “**ufw**” command with the “**deny**” option can be utilized for denying the traffic for port “**80**”:

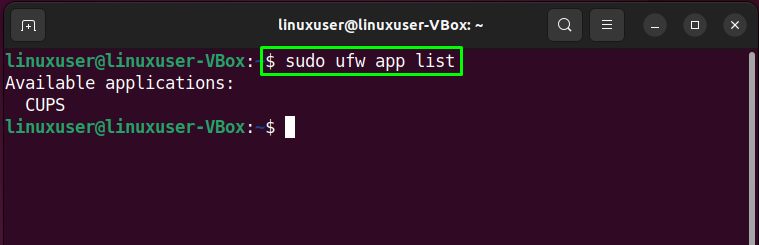
*$*sudo ufw deny 80



### ****Step 9: Check UFW list****

For the purpose of UFW application list, run the provided command:

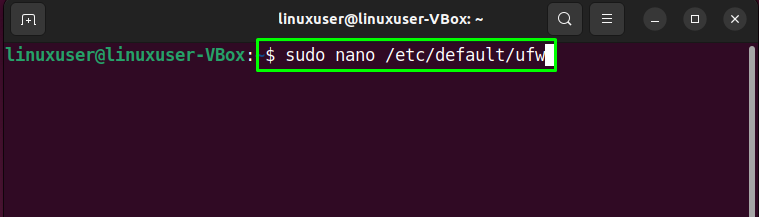
*$*sudo ufw app list



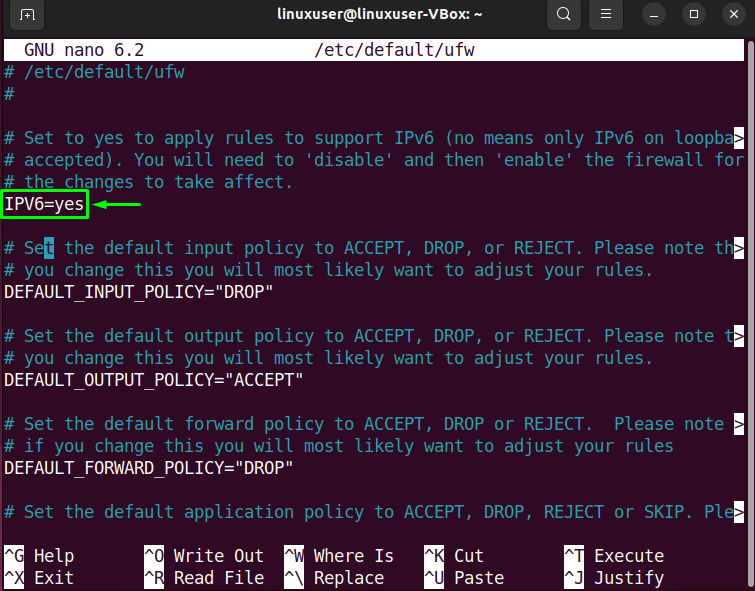
### ****Step 10: Open UFW configuration file****

We will now verify if the “**IPv6**” is working perfectly with UFW or not. To do so, open the UFW configuration file in the “**nano**” editor:

*$*sudo nano /etc/default/ufw



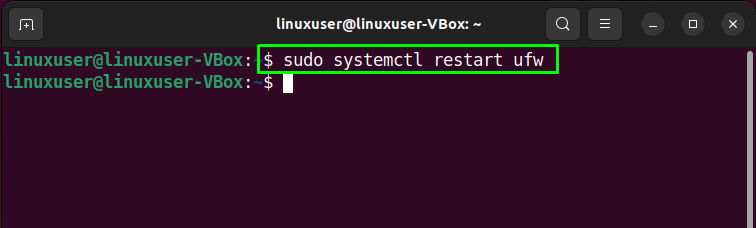
Then look for the line stating “**IPV6**” and assign “**yes**” as its value:



### ****Step 11: Restart UFW****

After performing the required changes, restart UFW by writing out the following command:

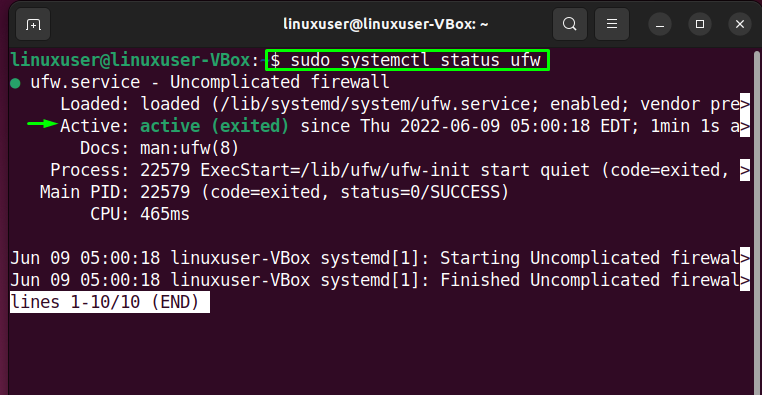
*$*sudo systemctl restart ufw



Lastly, check the status of UFW:

*$*sudo systemctl status ufw

The given output shows that UFW is currently active and running on our Ubuntu 22.04 system:

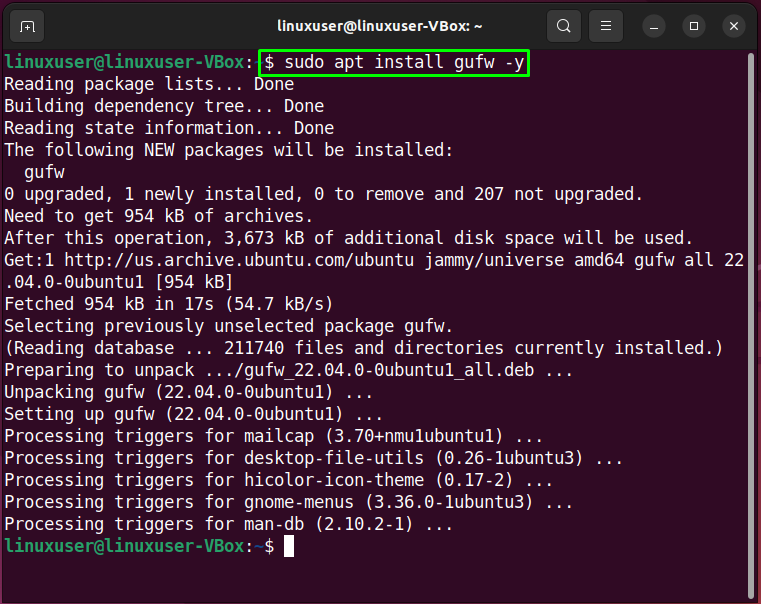


Now, let’s check out the method of configuring the advanced settings of UFW using the GUI application.

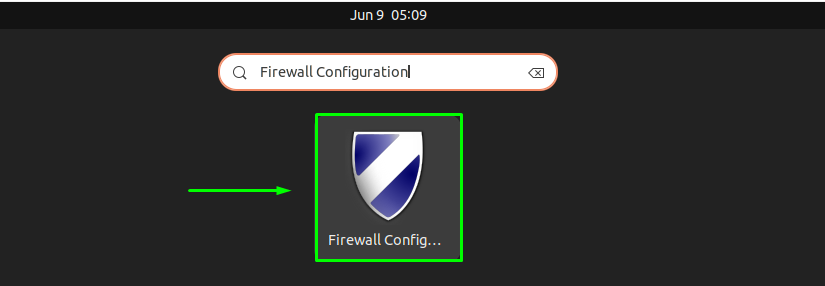
## ****How to perform Advanced UFW Configuration in Ubuntu 22.04 using GUI****

As Ubuntu 22.04 user, you can also perform the advanced configuration of UFW using its GUI application. Want to try it out? Hit “**CTRL+ALT+T**” and execute the provided command for the installation of **UFW GUI** application named “**gufw**”:

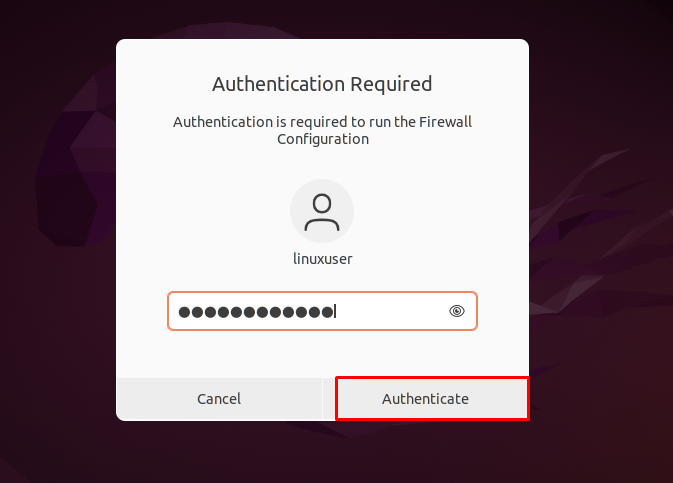
*$*sudo apt install gufw -y



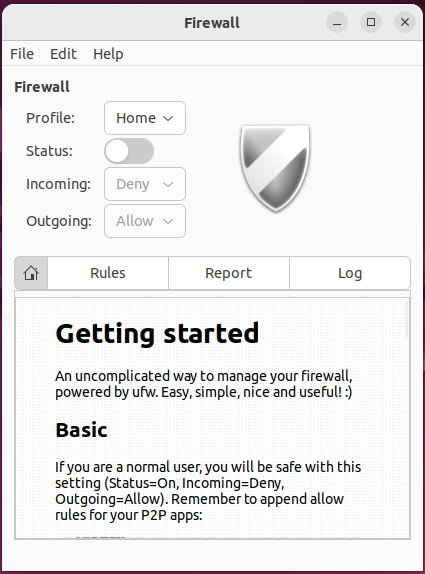
After successfully installing “**gufw**”, launch it on your system by searching “**Firewall Configuration**” in the “**Activities**” menu and open it from there:



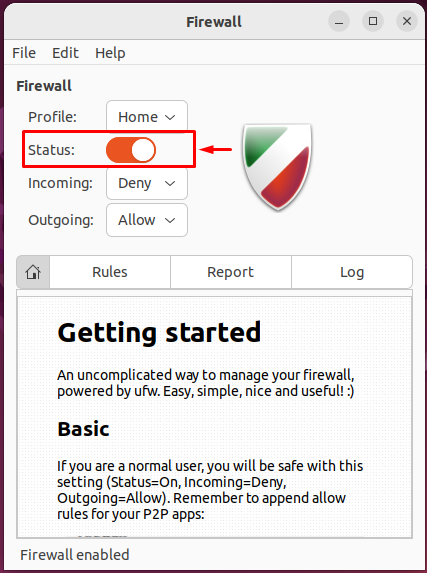
Then, specify your system password for the authentication purpose:



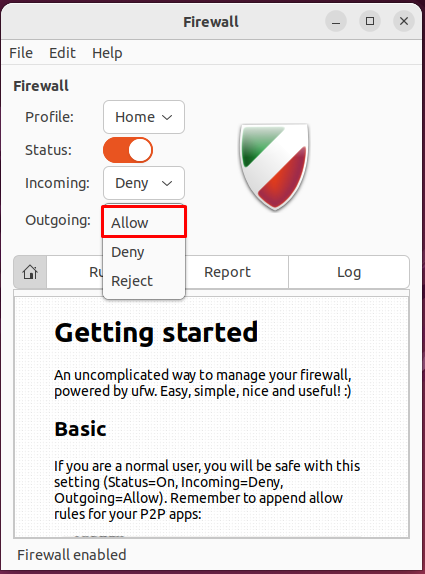
The main interface of “**gufw**” GUI UFW application will look like this:



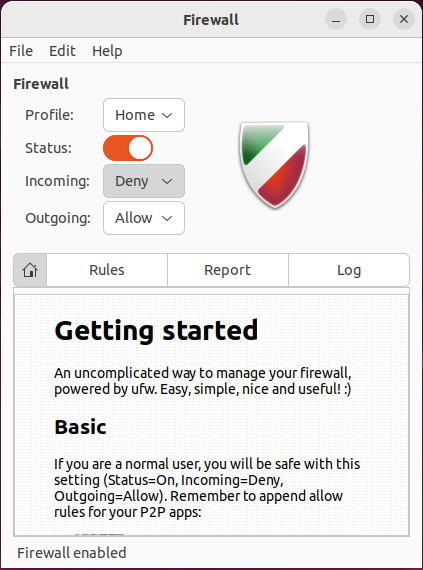
To enable the UFW Firewall, enable the highlighted “**Status**” toggle:



Similarly, select the required option for the “**Incoming**” and “**Outgoing**” connections:



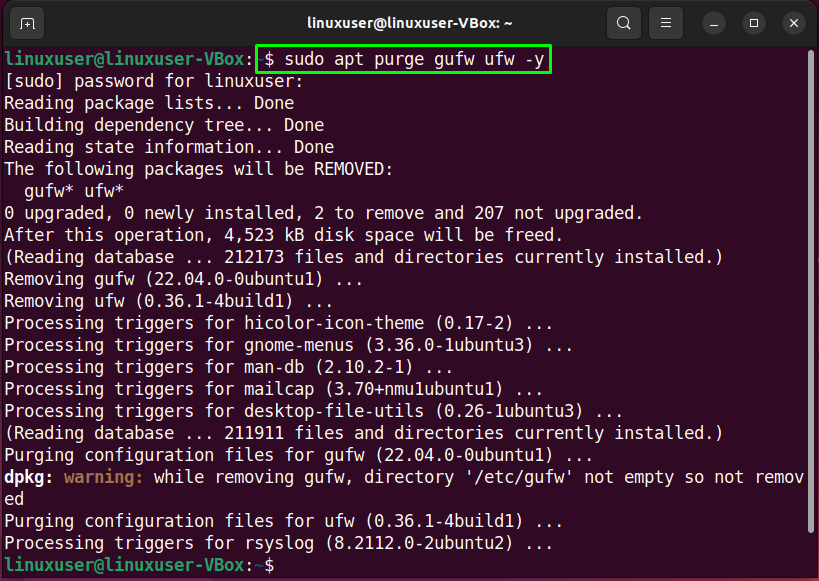
In our case, we have denied all incoming connection and allowed UFW to accept outgoing connections:



## ****How to uninstall UFW in Ubuntu 22.04****

For the uninstallation of UFW in 22.04, write out the provided command in the terminal:

*$*sudo apt purge gufw ufw -y



We have compiled the easiest method to perform advanced UFW configuration in Ubuntu 22.04 using the terminal and its GUI application.

**Configure rsyslog Server & Client in RHEL (CentOS 7,8).**

**Centralized Logging Server**

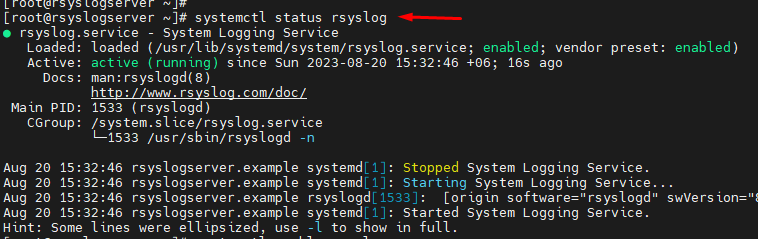
[root@rsyslogserver ~]# yum install rsyslog -y

[root@rsyslogserver ~]# systemctl restart rsyslog

[root@rsyslogserver ~]# systemctl start rsyslog

[root@rsyslogserver ~]# systemctl enable rsyslog

[root@rsyslogserver ~]# systemctl status rsyslog



[root@rsyslogserver ~]# vim /etc/rsyslog.conf

Find and uncomment following two directives.



[root@rsyslogserver ~]# systemctl restart rsyslog.service

[root@rsyslogserver ~]# firewall-cmd --permanent --add-port=514/tcp

[root@rsyslogserver ~]# firewall-cmd --permanent --add-port=514/udp

[root@rsyslogserver ~]# firewall-cmd --reload

[root@rsyslogserver ~]# firewall-cmd --list-all

[root@rsyslogserver ~]# tail /var/log/messages

[root@rsyslogserver ~]# cat /var/log/messages

**Configuring rsyslog Client.**

[root@rsyslogclient ~]# yum install rsyslog -y

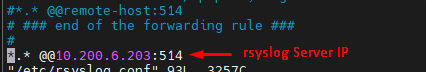
[root@rsyslogclient ~]# systemctl restart rsyslog

[root@rsyslogclient ~]# systemctl start rsyslog

[root@rsyslogclient ~]# systemctl enable rsyslog

[root@rsyslogclient ~]# systemctl status rsyslog

[root@rsyslogclient ~]# vim /etc/rsyslog.conf

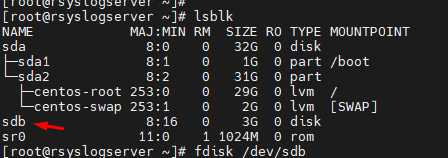


[root@rsyslogclient ~]# echo "\*.\* @@rsyslogserver.example:514" >> /etc/rsyslog.com

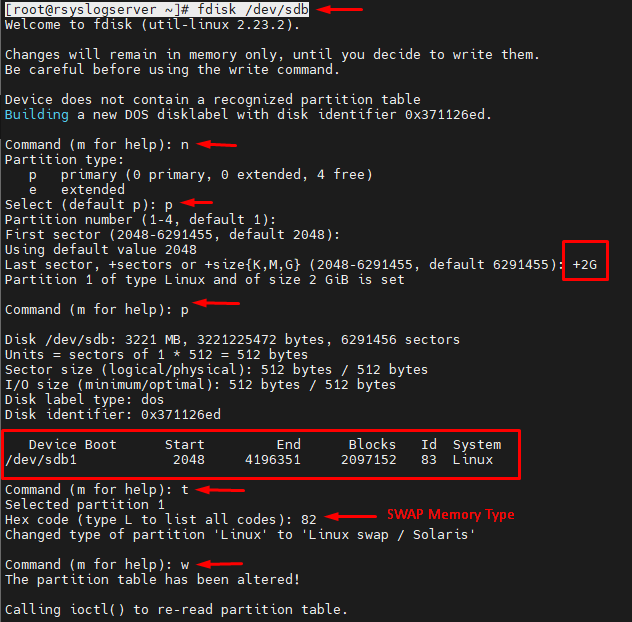
DNS Server

**Memory Management in Linux**

[root@rsyslogserver ~]# lsblk

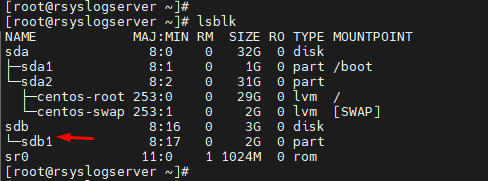


[root@rsyslogserver ~]# fdisk /dev/sdb

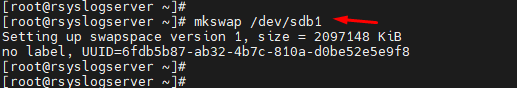


[root@rsyslogserver ~]# partprobe

[root@rsyslogserver ~]# lsblk



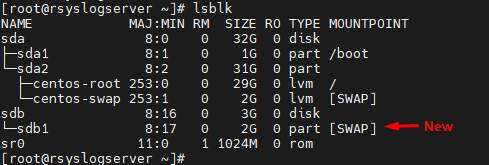
[root@rsyslogserver ~]# mkswap /dev/sdb1



[root@rsyslogserver ~]# cat /proc/swaps

[root@rsyslogserver ~]# swapon /dev/sdb1

[root@rsyslogserver ~]# lsblk



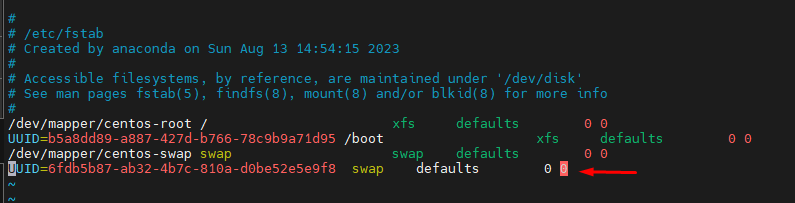
[root@rsyslogserver ~]# cat /proc/swaps



[root@rsyslogserver ~]# blkid /dev/sdb1



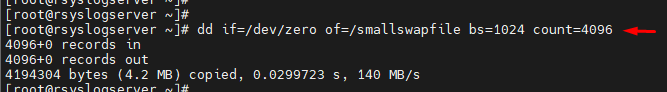
[root@rsyslogserver ~]# vim /etc/fstab



[root@rsyslogserver ~]# mount -a

**=================Directory SWAP Memory ================**

[root@rsyslogserver ~]# dd if=/dev/zero of=/smallswapfile bs=1024 count=4096



[root@rsyslogserver ~]# ls -l /smallswapfile

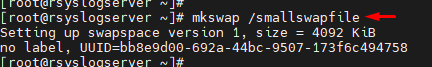
-rw-r--r-- 1 root root 4194304 Aug 20 17:19 /smallswapfile

[root@rsyslogserver ~]# chmod 0600 /smallswapfile

[root@rsyslogserver ~]# ls -l /smallswapfile

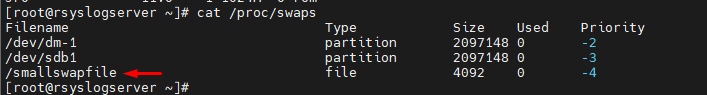
-rw------- 1 root root 4194304 Aug 20 17:19 /smallswapfile

[root@rsyslogserver ~]# mkswap /smallswapfile



[root@rsyslogserver ~]# swapon /smallswapfile

[root@rsyslogserver ~]# cat /proc/swaps



[root@rsyslogserver ~]# vim /etc/fstab

/smallswapfile swap defaults 0 0

[root@rsyslogserver ~]# vmstat

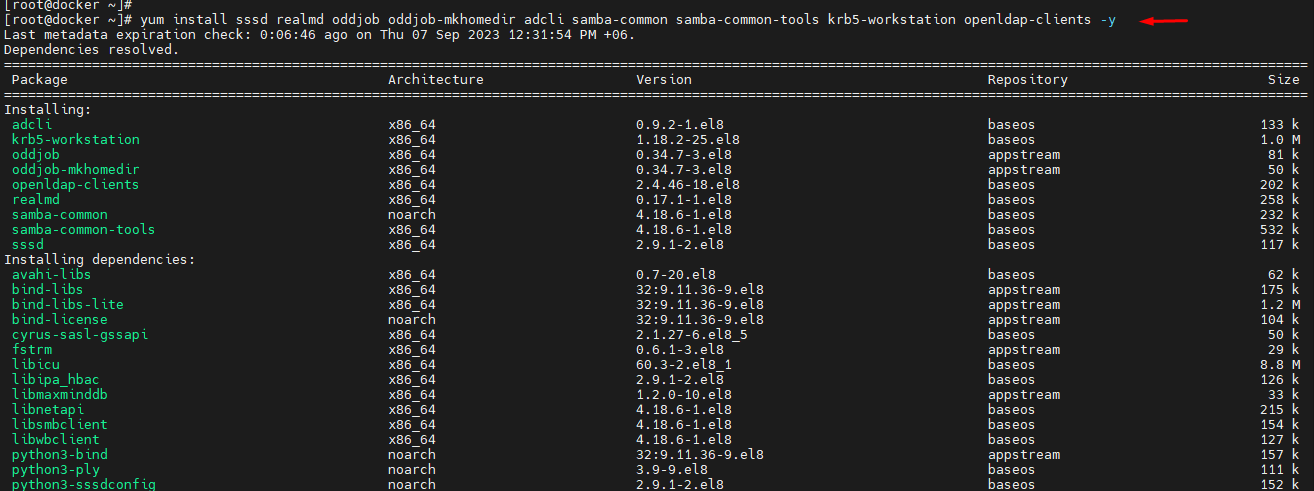
[root@rsyslogserver ~]# vmstat 2 100



=========================#######=========================

**Active Directory Windows Server 2016 Connect to Linux Server**

[root@docker ~]# yum install sssd realmd oddjob oddjob-mkhomedir adcli samba-common samba-common-tools krb5-workstation openldap-clients -y



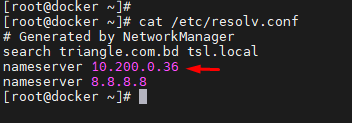
[root@docker ~]# cat /etc/resolv.conf

[root@docker ~]# vim /etc/resolv.conf

নোটঃ এখানে nameserver IP অ্যাড্রেস হচ্ছে ডোমেইন (AD) সার্ভারের আইপি।

[root@docker ~]# nslookup 10.200.0.36

36.0.200.10.in-addr.arpa name = dc.tsl.local.



[root@docker ~]# nslookup 10.200.6.12

\*\* server can't find 12.6.200.10.in-addr.arpa: NXDOMAIN

[root@docker ~]# nslookup docker.example.com

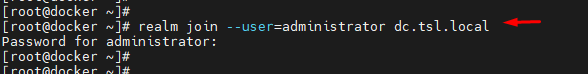
Server: 10.200.0.36

Address: 10.200.0.36#53

[root@docker ~]# realm join --user=administrator dc.tsl.local

Password for administrator: Admin@123

AD Administrator Password.



[root@docker ~]# id administrator

id: ‘administrator’: no such user

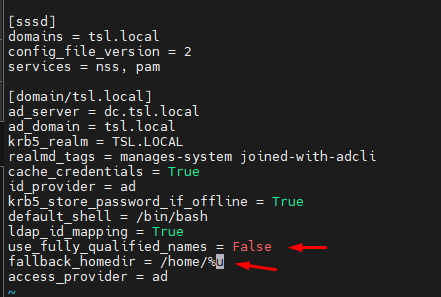
[root@docker ~]#

[root@docker ~]# id administrator@tsl.local

uid=1843600500(administrator@tsl.local)

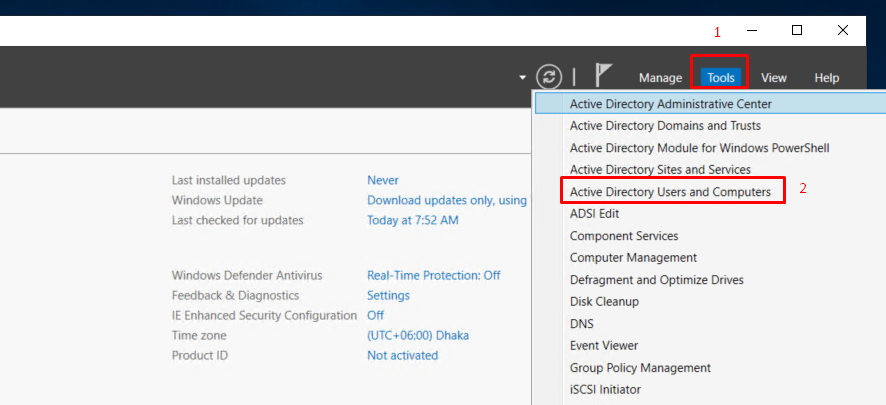
[root@docker ~]#

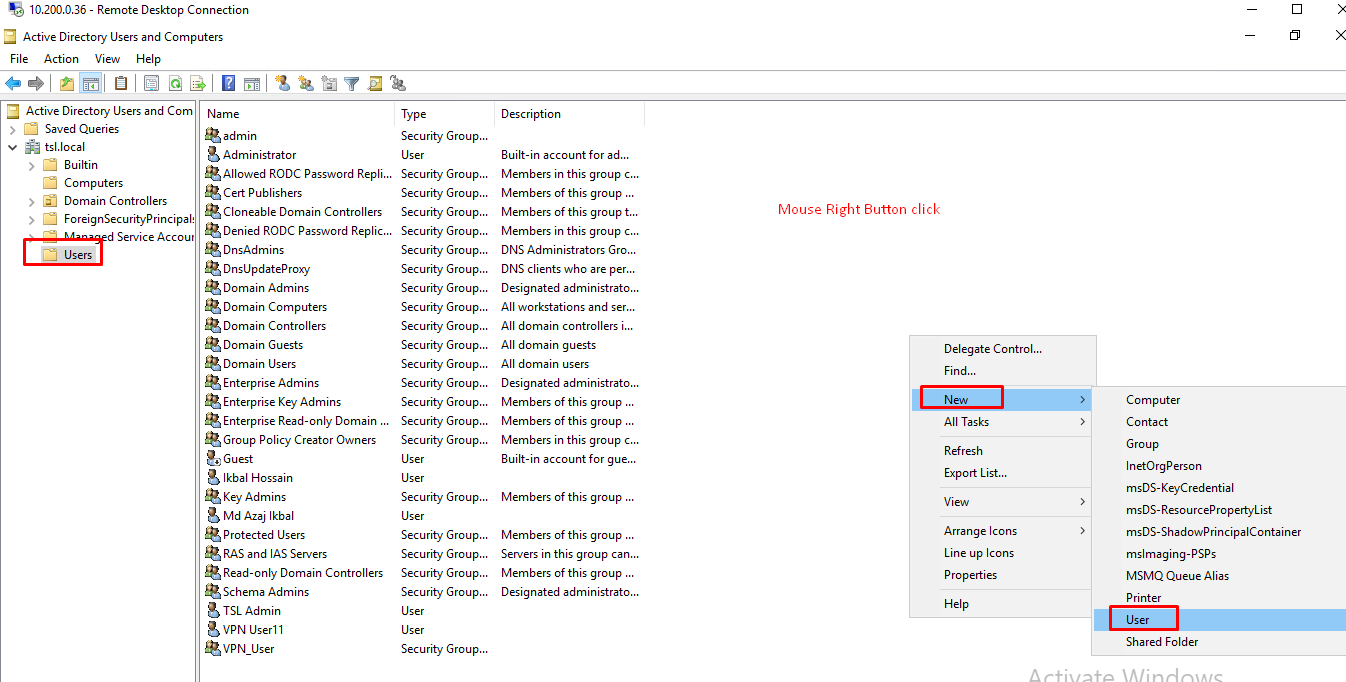
[root@docker ~]# vim /etc/sssd/sssd.conf



[root@docker ~]# systemctl restart sssd

**Active Directory Create**





Subdomain Create.

How To Join CentOS Linux To An Active Directory Domain:

-------------------------------------------------------

[root@samba ~]# yum install sssd realmd oddjob oddjob-mkhomedir adcli samba-common samba-common-tools krb5-workstation openldap-clients -y

[root@samba ~]# cat /etc/resolv.conf

search example.com

nameserver 172.25.11.254

নোটঃ এখানে nameserver IP অ্যাড্রেস হচ্ছে ডোমেইন (AD) সার্ভারের আইপি।

[root@samba ~]# nslookup dc1.example.com

Server: 172.25.11.254

Address: 172.25.11.254#53

Name: dc1.example.com

Address: 172.25.11.254

Join CentOS To Windows Domain:

------------------------------

[root@samba ~]# realm join --user=administrator example.com

Password for administrator: \*\*\*\*\*\*\*

Check your Samba Server on Windows SVR 2016:

--------------------------------------------

Tools -> Active Directory Users and Computers -> Computers

[root@samba ~]# id administrator

id: administrator: no such user

[root@samba ~]# id administrator@example.com

uid=1829600500(administrator@example.com)

[root@samba ~]# vim /etc/sssd/sssd.conf

use\_fully\_qualified\_names = True -> False

fallback\_homedir = /home/%u@%d -> /home/%u

:x

[root@samba ~]# systemctl restart sssd

[root@samba ~]# systemctl enable sssd

[root@samba ~]# id administrator

uid=1829600500(administrator)

[root@samba ~]# pdbedit -L

sadia:1001:

rakib:1003:

rafat:1002:

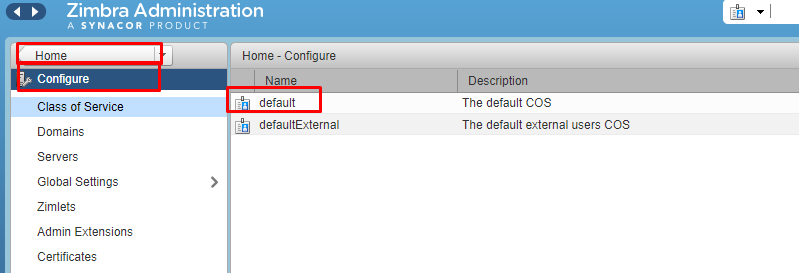
[root@samba ~]# smbpasswd -a samba1

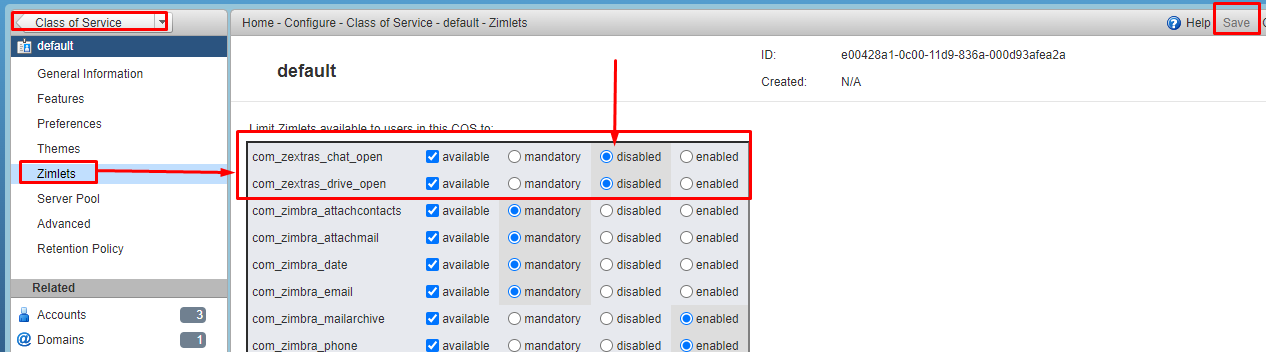
: \*\*\*\*\*

Leaving Domain:

--------------

[root@samba ~]# realm leave example.com





Form Add Command Group:

zmprov grr dl noc@triangle.com.bd usr nazmul.islam@triangle.com.bd sendAsDistList

**🡺Installing the NET-SNMP daemon on Debian Linux.**

1. Installation of the SNMP daemon

apt-get install snmpd -y

2. Create a backup of the SNMP configuration file

cp /etc/snmp/snmpd.conf /etc/snmp/snmpd.conf.orig

3. Adjust the configuration

# IPv4 setup

agentaddress udp:161

rocommunity public 0.0.0.0/0

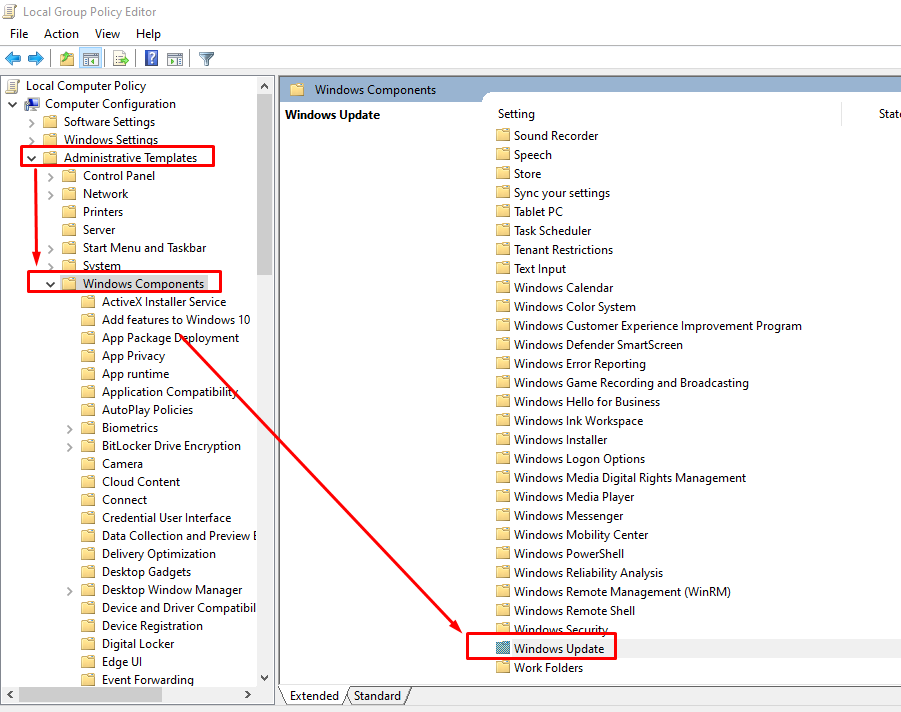
4.Save changes and restart the snmpd service

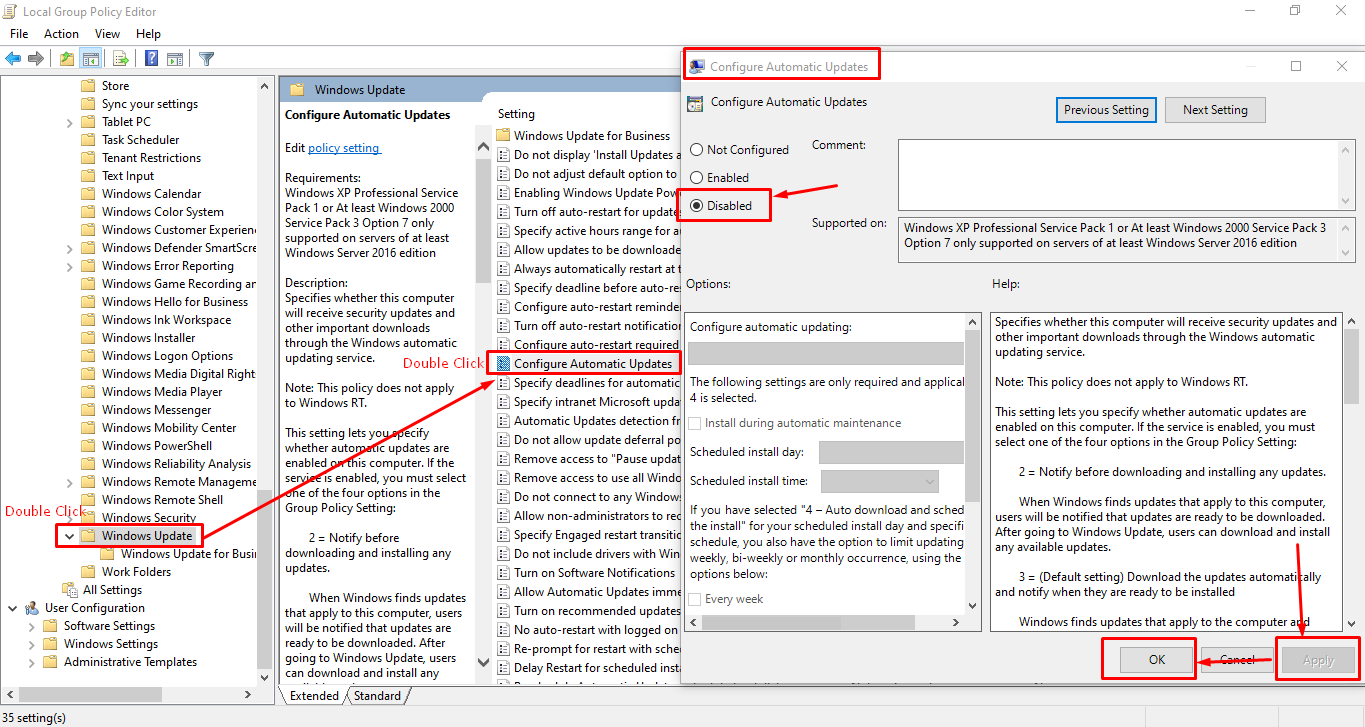
service snmpd restart

service snmpd enable

=============================================================================

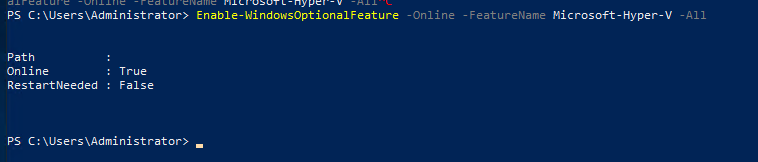
**Disable Windows 10 Automatic Updates**





**Enable virtualization without bios – Windows Server 2019 Hyper-V**

Command: Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V -All



Motherboard Model Check Command.

**[root@content ~]# dmidecode -t 2**

RAID bus controller Check Command.

**[root@content ~]# lspci | grep -i raid**

All Chipset Check Command.

**[root@content ~]# lspci | grep -i Chipset**

**[root@content ~]# sudo dmidecode -t 4**

**[root@content ~]# dmidecode -t processor | grep "Core Count"**

**Core Count: 8**

**Core Count: 8**

**[root@content ~]# cat /proc/cpuinfo |grep "processor"|wc -l**

**32**

**[root@content ~]# cat /proc/cpuinfo |grep "physical id"|sort |uniq|wc -l**

**2**

**[root@content ~]# dmidecode --type 17 //memory BUS and Other Info Check**

**[root@content ~]# cat /proc/meminfo**

## 1.2. Installing and enabling the web console

To access the RHEL 8 web console, first enable the cockpit.socket service.

Red Hat Enterprise Linux 8 includes the RHEL 8 web console installed by default in many installation variants. If this is not the case on your system, install the cockpit package before enabling the cockpit.socket service.

**Procedure**

1. If the web console is not installed by default on your installation variant, manually install the cockpit package:

# **yum install cockpit -y**

**Ubuntu Package Install.**

**#apt install cockpit -y**

1. Enable and start the cockpit.socket service, which runs a web server:

# **systemctl enable --now cockpit.socket**

1. If the web console was not installed by default on your installation variant and you are using a custom firewall profile, add the cockpit service to firewalld to open port 9090 in the firewall:
2. # **firewall-cmd --add-service=cockpit --permanent**

# **firewall-cmd --reload**

**Example:** [**https://localhost:9090**](https://localhost:9090)

## Disabling basic authentication in the web console

1. Open or create the cockpit.conf file in the /etc/cockpit/ directory in a text editor of your preference, for example:

**root@cockpit:~# vim /etc/cockpit/cockpit.conf**

# **vim cockpit.conf**

1. Add the following text:
2. **[basic]**

**action = none**

1. **Save the file.**
2. Restart the web console for changes to take effect.

# **systemctl try-restart cockpit**

## Adding a banner to the login page

You can set the web console to show a content of a banner file on the login screen.

**Procedure**

1. Open the /etc/issue.cockpit file in a text editor of your preference:

# **vim /etc/issue.cockpit**

1. Add the content you want to display as the banner to the file, for example:

Triangle Services Ltd. login banner for the RHEL web console login page.

You cannot include any macros in the file, but you can use line breaks and ASCII art.

1. **Save the file**.
2. Open the cockpit.conf file in the /etc/cockpit/ directory in a text editor of your preference, for example:

# **vim /etc/cockpit/cockpit.conf**

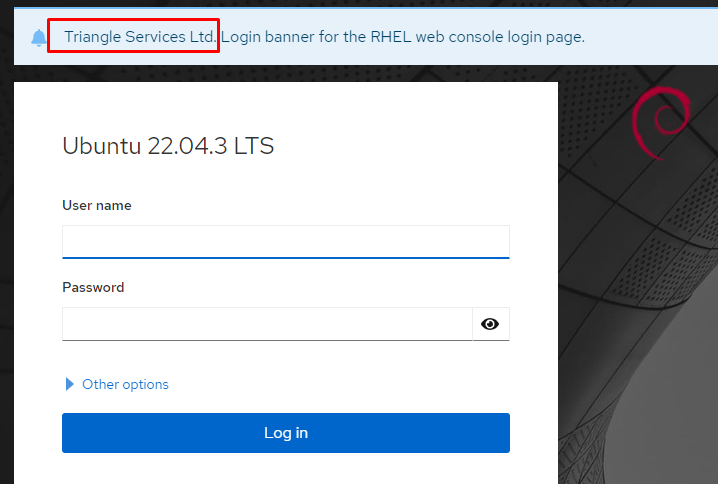
1. Add the following text to the file:

**[Session]**

**Banner=/etc/issue.cockpit**

1. **Save the file.**
2. Restart the web console for changes to take effect.

**root@cockpit:~#**  **systemctl try-restart cockpit**



## Configuring automatic idle lock in the web console

You can enable the automatic idle lock and set the idle timeout for your system through the web console interface.

**Procedure**

1. Open the cockpit.conf file in the /etc/cockpit/ directory in a text editor of your preference, for example:

# **vim /etc/cockpit/cockpit.conf**

1. Add the following text to the file:

**[Session]**

**IdleTimeout=<X>**

Substitute <X> with a number for a time period of your choice in minutes.

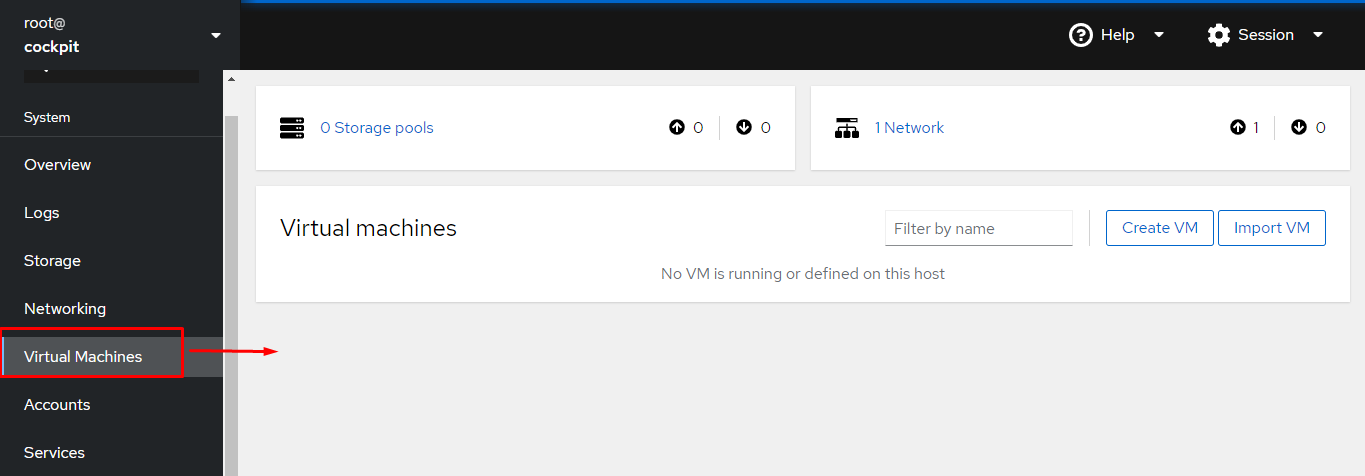
**Example: IdleTimeout=1**

1. **Save the file.**
2. Restart the web console for changes to take effect.

# **systemctl try-restart cockpit**

## Virtual machinesTools Install Web Console Protal.

**root@cockpit:~# apt install cockpit-machines -y**



https://github.com/optimans/cockpit-zfs-manager

Control+X+E

$ git clone https://github.com/optimans/cockpit-zfs-manager.git

$ sudo cp -r cockpit-zfs-manager/zfs /usr/share/cockpit

