Ctrl+alt+F3 for console

Ctrl+alt+F7

su

enter the root password

nano /etc/apt/sources.list

place # sign before cdrom

ctrl+o to save, ctrl+x to exit

apt update

apt install linux-headers-amd64 gcc make perl

sudo nano /etc/fstab

change nanoauto to exec

umount /media/cdrom0

mount –a

umount /media/cdrom0

to add user

su –

password -----

root@koha:~# adduser rabeya

Adding user `rabeya' ...

Adding new group `rabeya' (1001) ...

Adding new user `rabeya' (1001) with group `rabeya' ...

Creating home directory `/home/rabeya' ...

Copying files from `/etc/skel' ...

New password:

Retype new password:

passwd: password updated successfully

Changing the user information for rabeya

Enter the new value, or press ENTER for the default

Full Name []:

Room Number []:

Work Phone []:

Home Phone []:

Other []:

Is the information correct? [Y/n] y

root@koha:~# usermod -aG sudo masum

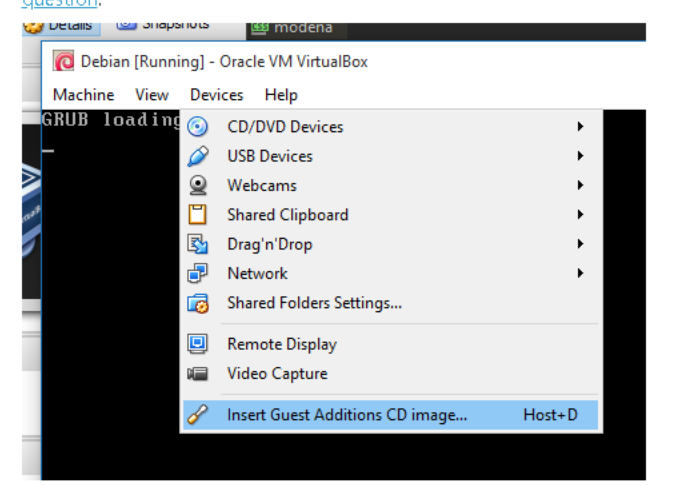
root@koha:~# exit

logout

another perfect option

# apt update

# apt install build-essential dkms



sudo apt update

sudo apt install build-essential dkms linux-headers-$(uname -r)

sudo mkdir -p /mnt/cdrom

sudo mount /dev/cdrom /mnt/cdrom

cd /mnt/cdrom

sudo sh ./VBoxLinuxAdditions.run --nox11

sudo shutdown -r now

lsmod | grep vboxguest

sudo reboot

lsmod

lsmod | grep vbox

ls -la /media

cd /media/sf\_sandbox i.e shared folder name

masum@dspace:~$ sudo nano /etc/NetworkManager/NetworkManager.conf

logout/restart the system and back in to take the change into effect.

sudo touch /var/myfile

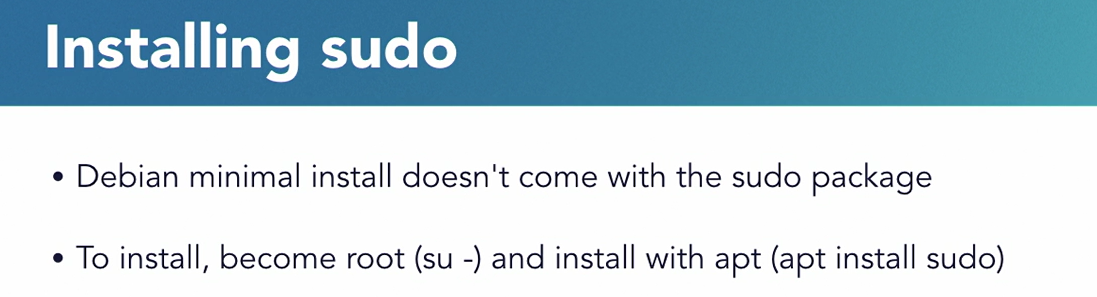
[sudo] password for masum:

I don’t need the file just for practice I created therefore I delete with the following command

*This will not take effect until restart the system*

sudo rm /var/myfile

no password required



masum@koha:~$ sudo usermod -L root

masum@koha:~$ sudo visudo

regular user don’t have the sbin directory included their path variable but using sudo we get a different PATH variable temporarily because of the secure path variable in the /etc is sudoers file, so If you are working as a regular user and type a command you would like to administer the system instead of saying you do not have permission you may run some command that say not found, and that’s why the regular user path doesn’t include a location where some system management tools are kept. But when we switch to using super user its included.

Leave with ctrl + x

isudo: /etc/sudoers.tmp unchanged

working with users is a task you may find yourself meet in *to do* as you administer your system.

In a linux system every process is run has a process id or PID number

With these number we can change the processes and their priority

To see process Type

Top

X to highlight the current processes taking more cpu

B to we can highlit colmn cpu

< > greater than and less than sign we can move from column to column

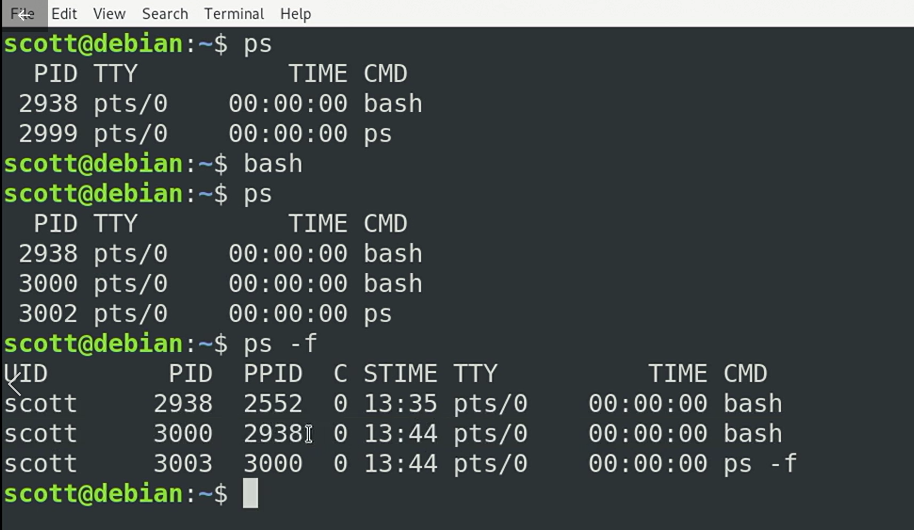
Capital R reverses the sort order of the row

K key to choose a process id to kill lower case r key to choose a process id to renice

Q for quit

Typed bash to create another process

Ps command shows the processes within this shell and ps –f to see the parent process id



As parent process id ppid correspond to the original shell pid

ps -fh to see the processes hierarchically

ps –eH for more information

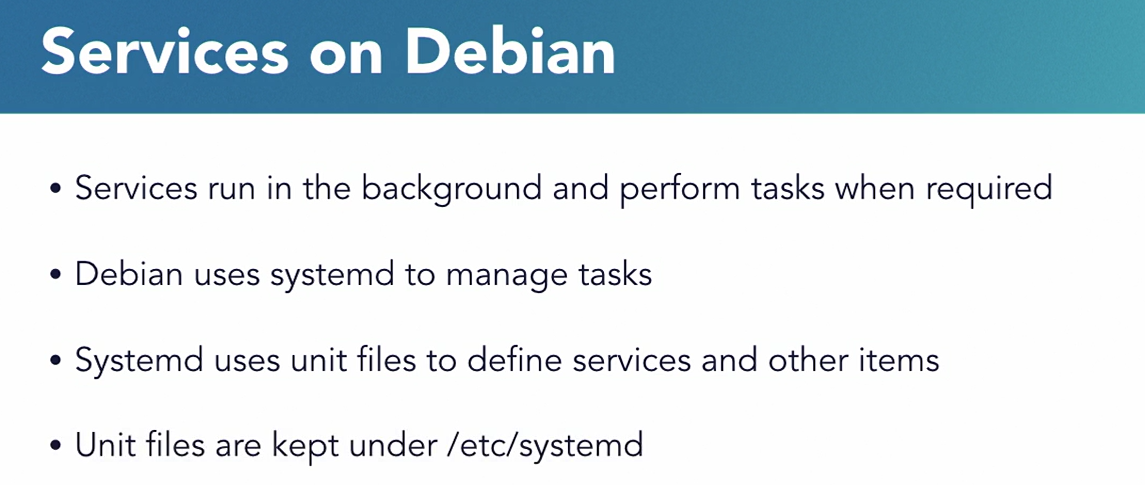
Kill –l

To see wha the processes can be killed and in the following image describes how processes created and can be killed



Process id show up in many places it needs to know what they are and where those can be used for.

*Managing Services:*

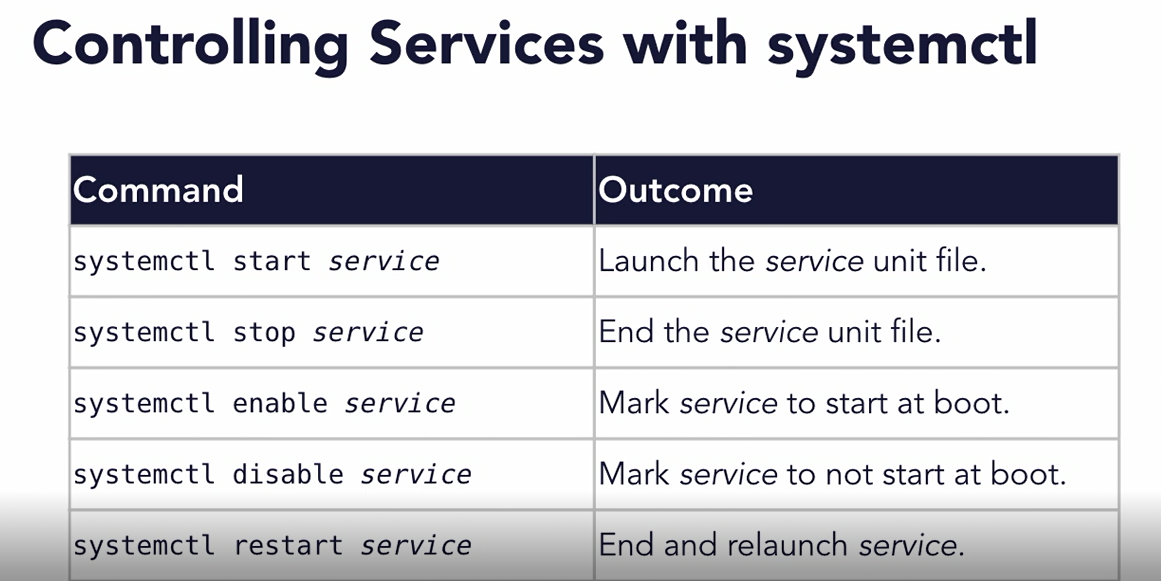


systemctl

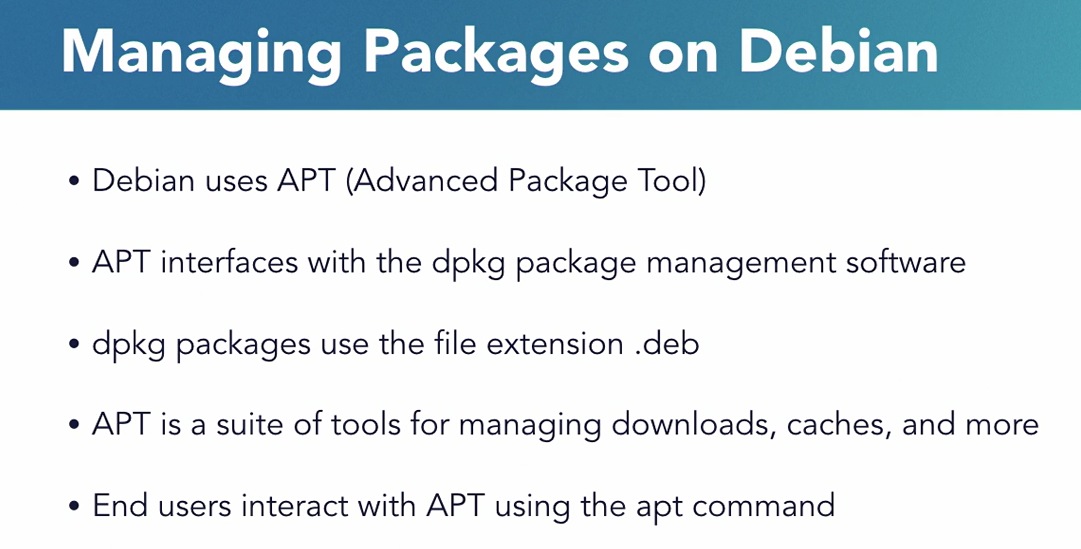
systemctl –t service

we can see any particular service by typing

systemctl status rsyslog.service



***Managing software with apt***



Apt can be used to work with packages are kept locally on the system

masum@koha:~$ ls /etc/apt

apt.conf.d listchanges.conf sources.list sources.list.d

auth.conf.d preferences.d sources.list~ trusted.gpg.d

masum@koha:~$ apt list tree this performs the search of the package name

Listing... Done

tree/stable 1.8.0-1 amd64

masum@koha:~$ apt search tree

masum@koha:~$ apt show tree

the we see the package name version maintainer size etc

masum@koha:~$ sudo apt install tree

masum@koha:~$ tree -d /etc

masum@koha:~$ sudo apt upgrade

Reading package lists... Done

Building dependency tree …..

Apt list --installed | less

sudo apt install unattended-upgrades

sudo dpkg-reconfigure unatteneded –upgrades

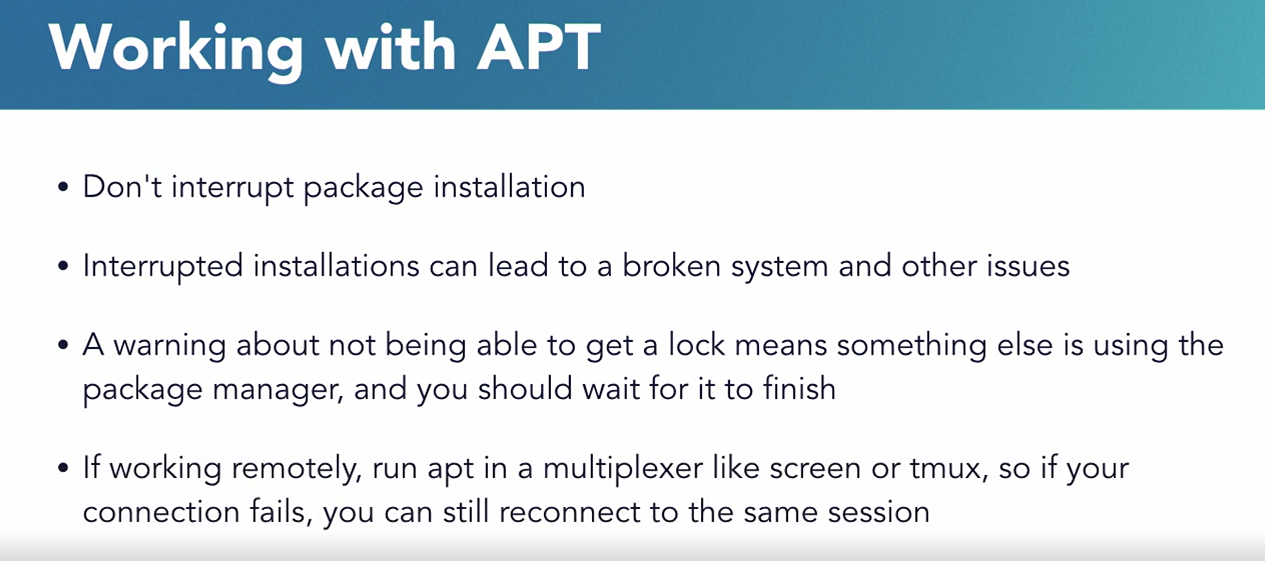
masum@koha:~$ sudo dpkg-reconfigure unatteneded-upgrades

[sudo] password for masum:

dpkg-query: package 'unatteneded-upgrades' is not installed and no information is available

Use dpkg --info (= dpkg-deb --info) to examine archive files.

/usr/sbin/dpkg-reconfigure: unatteneded-upgrades is not installed



Take sometime to explore the repository

**Apparmor**

****

unconfined programs are not subject to the apparmor controls

we can check it with

masum@koha:~$ sudo aa-status

apparmor module is loaded.

15 profiles are loaded.

13 profiles are in enforce mode………..

Apparmor profile is kept in the etc directory

masum@koha:~$ ls /etc/apparmor.d/

abstractions usr.bin.evince

disable usr.bin.man

force-complain usr.lib.libreoffice.program.oosplash

local usr.lib.libreoffice.program.senddoc

nvidia\_modprobe usr.lib.libreoffice.program.soffice.bin

tunables usr.lib.libreoffice.program.xpdfimport

sudo nano /etc/apparmor.d/usr.bin.evince

below the textfile we would see root access and rw to directories

lets make a change here I will add a line to deny access to the downloads folder to the user home folder

deny @{HOME}/Downloads/ r,

this change doesn’t make lot of effects administratively but this will work for the demonstration what the effect is

save this

and run

sudo systemctl restart apparmor.service

then run

evince

close the app for now

run

dmesg

dmesg: read kernel buffer failed: operation not permitted

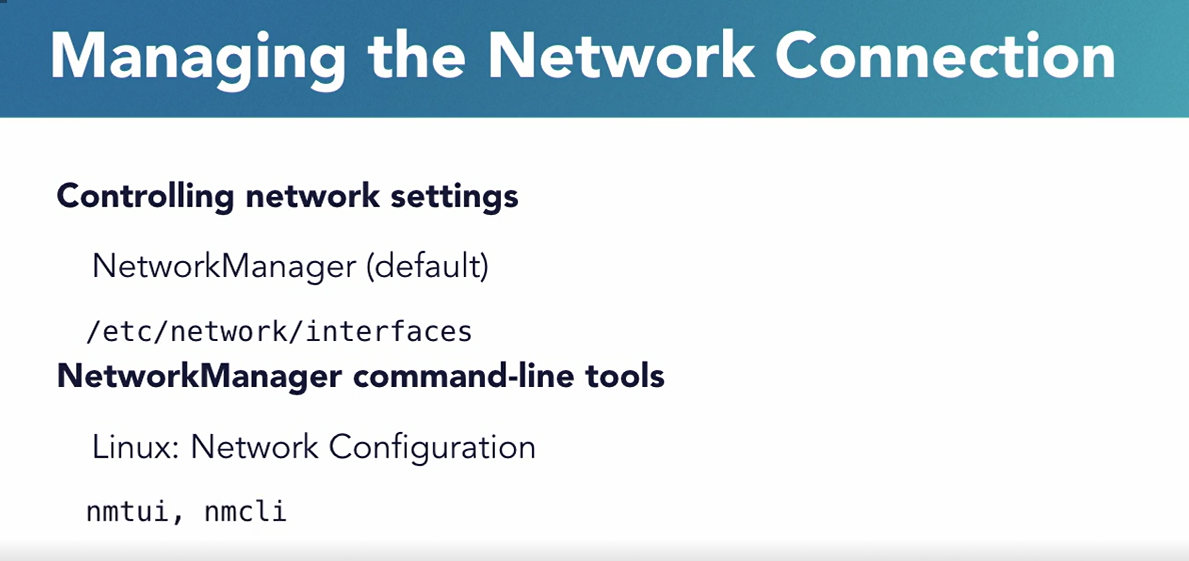
sudo dmesg to see the system log there will be errors to browse into directory where the app wasn’t allowed to read.

Undo the things following same way

Manipulating apparmor profile probabely you won’t need to do on daily basis.

And entirely likely you will never need to edit this profile at all.

But it was to show the basic of apparmor





**Debian** apt-get install network-manager

**Ubuntu** apt-get install network-manager

**Alpine** apk add networkmanager

**Arch Linux** pacman -S networkmanager

**Kali Linux** apt-get install network-manager

**Fedora** dnf install NetworkManager-tui-1

**Raspbian** apt-get install network-manager

**Docker** docker run cmd.cat/nmtui nmtui

masum@dspace:~$ sudo systemctl restart network-manager

masum@dspace:~$ sudo systemctl status NetworkManager

masum@koha:~$ sudo nmtui

DNS configuration:

servers: 192.168.1.1

interface: enp0s3

=== [id] ===

[NM property description]

A human readable unique identifier for the connection, like "Work Wi-Fi" or "T-Mobile 3G".

nmcli> set connection.id MyEthernet [ failed to save]

masum@dspace:~$ nmcli c

NAME UUID TYPE DEVICE

Wired connection 1 cecf0a7c-69bd-3671-871d-5441e1cc3302 ethernet enp0s3

Uuid cecf0a7c-69bd-3671-871d-5441e1cc3302

masum@dspace:~$ nmcli c

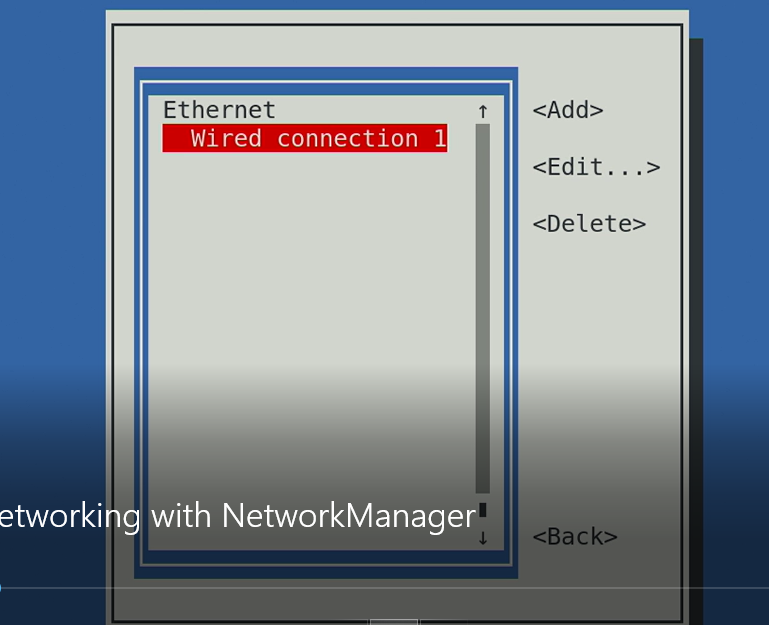
NAME UUID TYPE DEVICE

Wired connection 1 cecf0a7c-69bd-3671-871d-5441e1cc3302 ethernet enp0s3

masum@dspace:~$ sudo nmcli con up cecf0a7c-69bd-3671-871d-5441e1cc3302

[sudo] password for masum:

Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/2)



Do accordingly

masum@koha:~$ nmcli it gives us information about active network connection

the object at the bottom give us the opportunity to work with devices connections and so on

Consult nmcli(1) and nmcli-examples(5) manual pages for complete usage details.

masum@dspace:~$ nmcli device show

GENERAL.DEVICE: enp0s3

GENERAL.TYPE: ethernet

GENERAL.HWADDR: 08:00:27:AF:55:D6

GENERAL.MTU: 1500

GENERAL.STATE: 10 (unmanaged)

GENERAL.CONNECTION: --

GENERAL.DEVICE: enp0s3

GENERAL.TYPE: ethernet

GENERAL.HWADDR: 08:00:27:AF:55:D6

GENERAL.MTU: 1500

GENERAL.STATE: 10 (unmanaged)

GENERAL.CONNECTION: --

GENERAL.CON-PATH: --

WIRED-PROPERTIES.CARRIER: on

IP4.ADDRESS[1]: 10.0.2.15/24

IP4.GATEWAY: 10.0.2.2

IP4.ROUTE[1]: dst = 0.0.0.0/0, nh = 10.0.2.2, mt = 0

IP4.ROUTE[2]: dst = 10.0.2.0/24, nh = 0.0.0.0, mt = 0

IP4.ROUTE[3]: dst = 169.254.0.0/16, nh = 0.0.0.0, mt =

IP6.ADDRESS[1]: fe80::a00:27ff:feaf:55d6/64

IP6.GATEWAY: --

IP6.ROUTE[1]: dst = fe80::/64, nh = ::, mt = 256

GENERAL.DEVICE: lo

GENERAL.TYPE: loopback

GENERAL.HWADDR: 00:00:00:00:00:00

GENERAL.MTU: 65536

GENERAL.STATE: 10 (unmanaged)

GENERAL.CONNECTION: --



A connection is a particular settings applied to particular device

To work with we use c short for connection

masum@koha:~$ nmcli c

NAME UUID TYPE DEVICE

Wired connection 1 89a53dc0-e6b0-4c93-bcfa-b757e78b0124 ethernet enp0s3

Or

masum@koha:~$ nmcli connection

NAME UUID TYPE DEVICE

Wired connection 1 89a53dc0-e6b0-4c93-bcfa-b757e78b0124 ethernet enp0s3

masum@koha:~$ nmcli d

DEVICE TYPE STATE CONNECTION

enp0s3 ethernet connected Wired connection 1

lo loopback unmanaged --

to see what we can do with the connection

masum@koha:~$ nmcli c –help

nmcli con e

print connection

print ipv4

connection.id: ethernet

connection.uuid: 5beabf05-27bb-4eee-9f87-eeaa43901f1a

connection.stable-id: --

connection.type: 802-3-ethernet

connection.interface-name: --

connection.autoconnect: yes

connection.autoconnect-priority: 0

connection.autoconnect-retries: -1 (default)

connection.multi-connect: 0 (default)

connection.auth-retries: -1

connection.timestamp: 0

connection.read-only: no

connection.permissions: --

connection.zone: --

connection.master: --

connection.slave-type: --

connection.autoconnect-slaves: -1 (default)

connection.secondaries: --

connection.gateway-ping-timeout: 0

connection.metered: unknown

connection.lldp: default

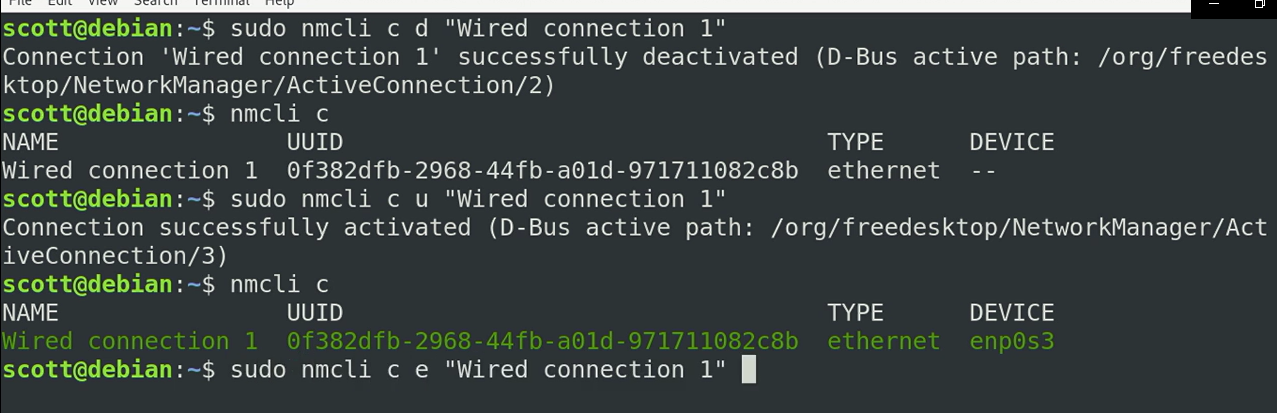
connection.mdns: -1 (default)

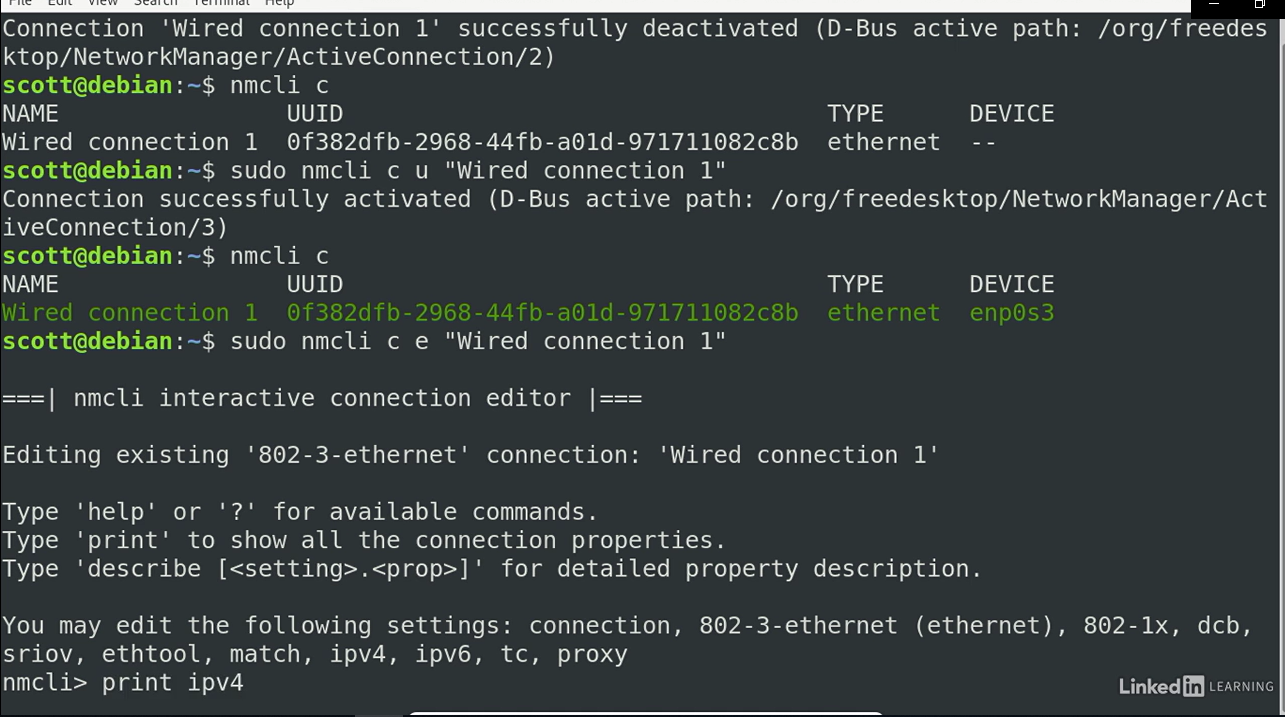
connection.llmnr: -1 (default)

I can set it up and down to enable and disable it press q to quit

192.168.1.102

192.168.1.1





Nmcli > print ipv4 show ipv4 information

Nmcli > connection shows the information about the connection itself

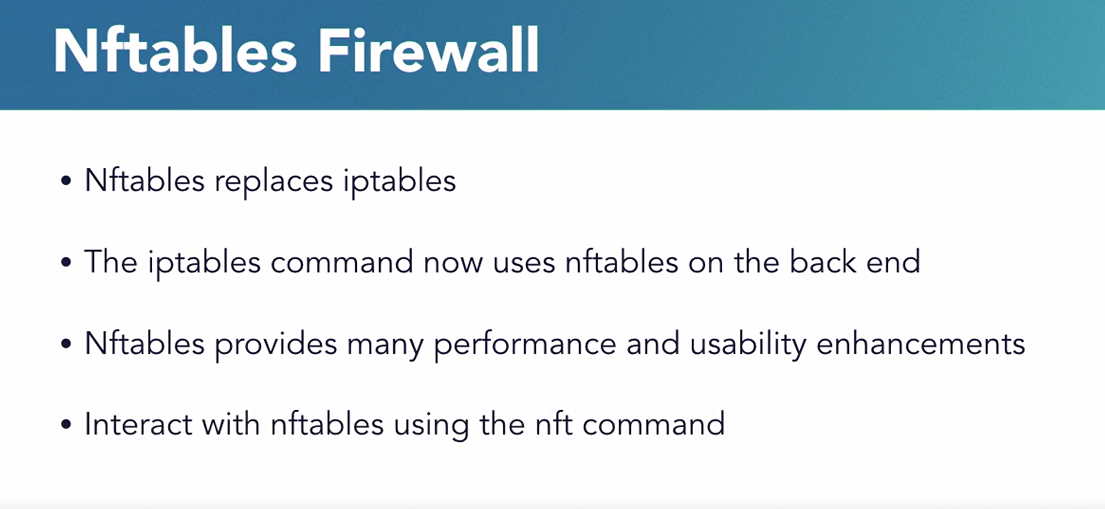
Nmcli> describe connection or describe ipv4 will give you listing what each item represents

We can remove and delete to modify and when we are done we type simply

Nmcli> save

Nmcli> quit

masum@koha:~$ nmcli --help



We also install software to get the nft command

masum@koha:~$ sudo apt install nftables

masum@koha:~$ sudo systemctl enable nftables

the output is

Created symlink /etc/systemd/system/sysinit.target.wants/nftables.service → /lib/systemd/system/nftables.service.

masum@koha:~$ sudo systemctl start nftables

like ip tables nftables operates using tables change and rules where tables represent collection of change and and changes are sets or rules that act as filters for packets that evaluated by a given chain. Each rule is written as filter for particular types of traffic and every rules ends with a verdict or the actions to be taken when the rules matches packet

masum@koha:~$ sudo nft list tables

table inet filter

masum@koha:~$ sudo nft list chains

table inet filter {

chain input {

type filter hook input priority 0; policy accept;

}

chain forward {

type filter hook forward priority 0; policy accept;

}

chain output {

type filter hook output priority 0; policy accept;

}

}

masum@koha:~$

out of the box we have three chains for input traffic forwarding traffic and output traffic and they are all configured to accept any traffic rather they don’t hav any rules to filter traffic their default action is to accept everything so the lockdown access we need to take some change. It is deny some traffic and set what we allow to pass here we focus on the incoming or the input chain which coming into our system from somewhere else in the network, we can see that in with the command

masum@koha:~$ sudo nft list chain inet filter input

table inet filter {

chain input {

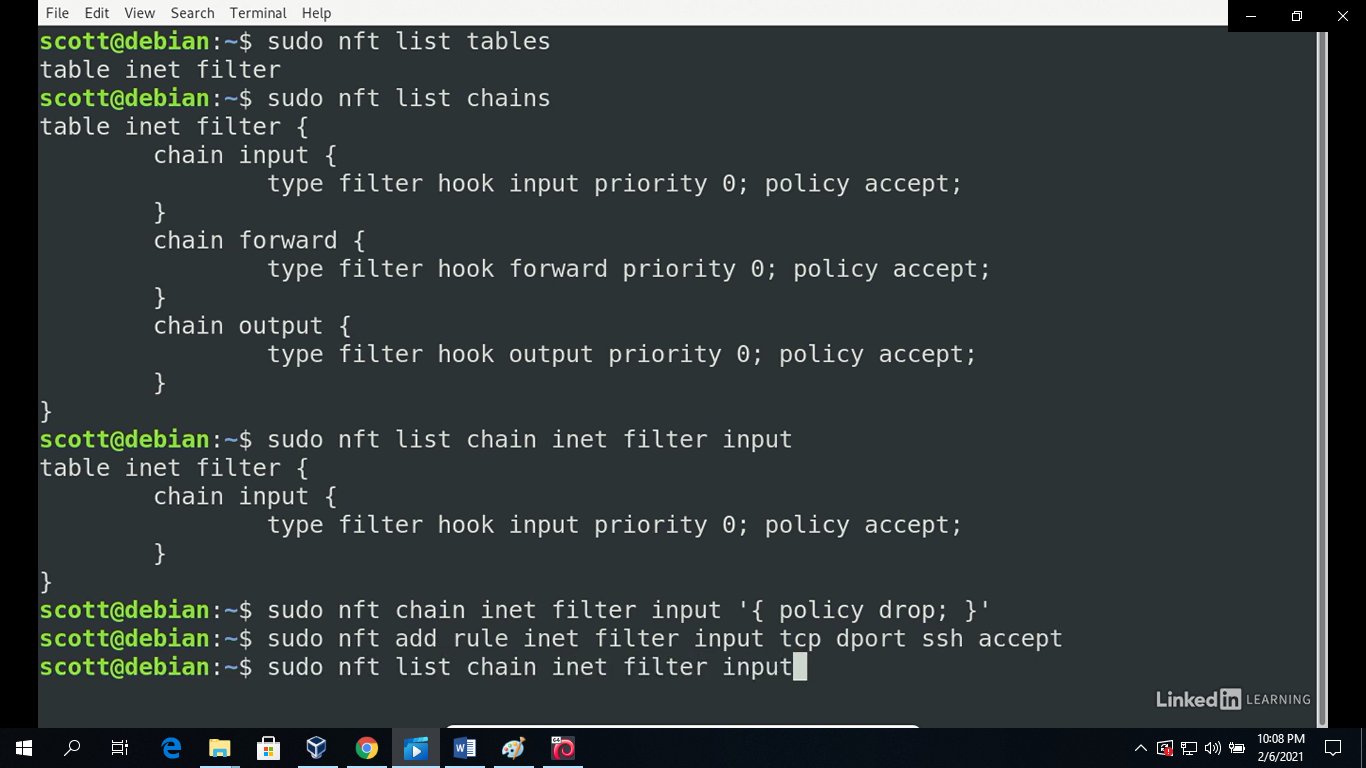
type filter hook input priority 0; policy accept;

}

}

We can add a rule by two ways either by editing the configuration file /etc/nftables.conf

Or using command line tools to work with chain individually



masum@koha:~$ sudo nft chain inet filter input '{policy drop;}'

masum@koha:~$ sudo nft add rule inet filter input tcp dport ssh accept

masum@koha:~$ sudo nft list chain inet filter input

table inet filter {

chain input {

type filter hook input priority 0; policy drop;

tcp dport ssh accept

}

}

One thing to be aware of here is the this interactive commands set the current states of the firewall but if the firewall restarts when it happens when the system is reboot this changes go away

So to make them permanent we need to add them in the configuration file so they autoloaded when the firewall starts

That configuration file is at

#!/usr/sbin/nft -f

flush ruleset

masum@koha:~$ sudo nano /etc/nftables.conf

table inet filter {

chain input {

type filter hook input priority 0;

tcp dport ssh accept

ct state established, related accept

ip protocol icmp accept

iif lo accept

}

chain forward {

type filter hook forward priority 0;

}

masum@koha:~$ sudo systemctl restart nftables

masum@dspace:~$ ss -tl

State Recv-Q Send-Q Local Address:Port Peer Address:Port

LISTEN 0 128 0.0.0.0:ssh 0.0.0.0:\*

LISTEN 0 128 [::]:ssh [::]:\*

a debian system keeps logs of system activity that you can browse for troubleshooting ti keep watch whats going on most of these are kept in the /var/log directory

masum@koha:~$ cd /var/log

masum@koha:/var/log$

masum@koha:/var/log$ ls

alternatives.log fontconfig.log private

apt gdm3 speech-dispatcher

auth.log installer syslog

boot.log kern.log unattended-upgrades

btmp lastlog user.log

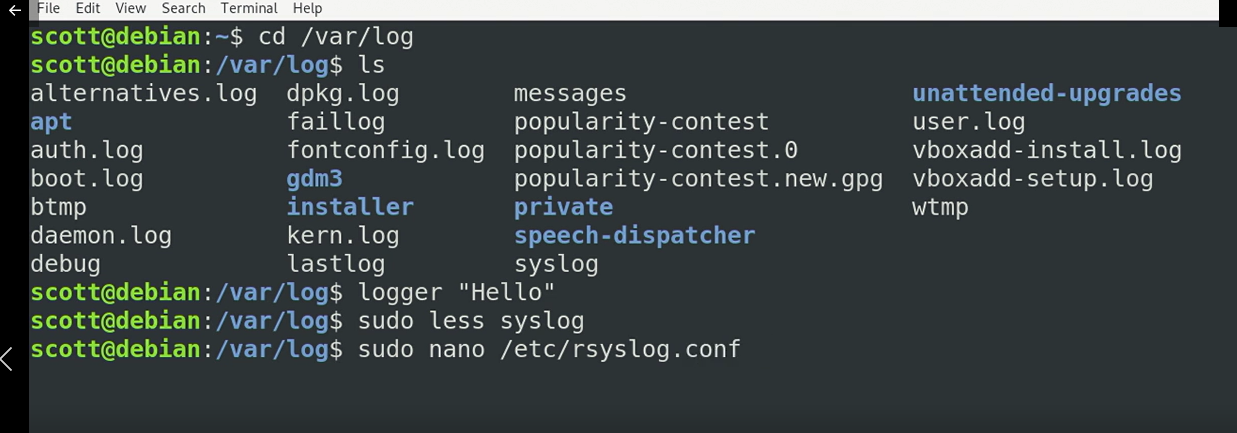
daemon.log messages vboxadd-install.log

debug popularity-contest vboxadd-setup.log

dpkg.log popularity-contest.0 wtmp

faillog popularity-contest.new.gpg

some of the interesting log here is the syslog which is the overall system log kern.log which is the kernel log and auth log which is the log of authentication that have occurred for the most part we focus on system log because it includes the information and for kernel log and other sources many programs and services write to this log you can add entries to the line from the command line with a logger command



Sudo syslog for brose through the files wih the up and down arrow keys forward and backward with f and b we move forward here wand will find our logger input hello

Its worth mentioning the system logger syslog sends log to our remote server and accepts log from other system

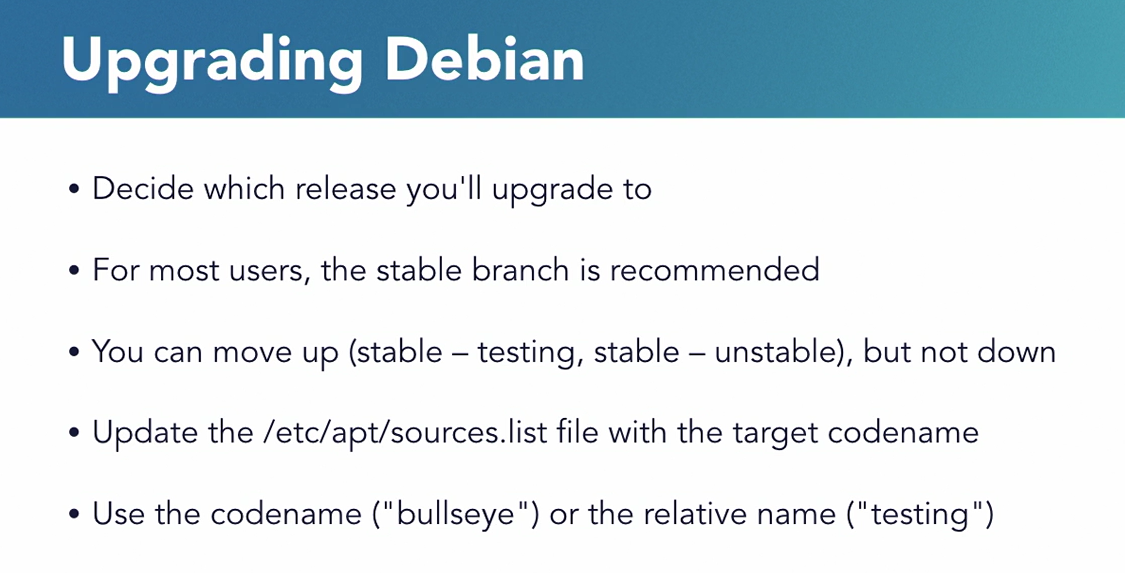
Auth.log file stores information about authentication

masum@koha:/var/log$ sudo tail auth.log

masum@koha:/var/log$ sudo tail kern.log

kern.log shows information of the kernel that is system startup information hardware information and changes to the network interface we can also get same information using dmesg command

logs are great place to start when you are troubleshooting



Don’t do this it will break the system with less core

You cant easily downgrade your system without reinstall it

Sudo nano /etc/apt/source/list

Ctrl + \ for find and replace buster with bullseye

sudo apt update

sudo apt upgrade

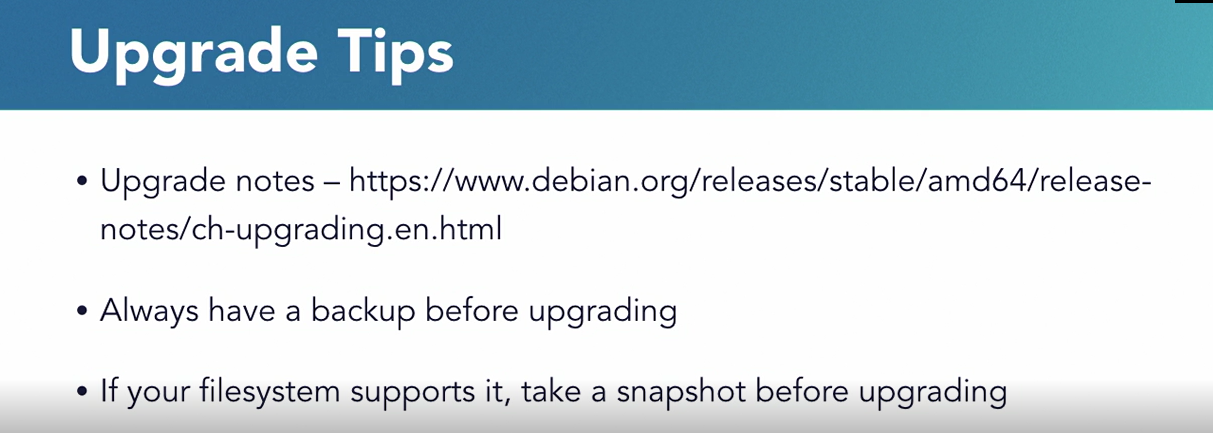
press q to quit select ok and choose yes

again select ok and choose yes

sudo apt full-upgrade to move my system to the new release

sudo shutdown –r now

cat /etc/or-release its always ideal to move one stable release t the next



masum@dspace:~$ cat /etc/os-release

Later gnome desktop

Sudo tasksel

Xfce lightweight desktop

masum@debian:~$ sudo nano /etc/NetworkManager/NetworkManager.conf

[main]

plugins=ifupdown,keyfile

[ifupdown]

managed=false

sudo nano /etc/network/interfaces

auto enp0s3

or

allow-hotplug enp0s3

iface enp0s3 inet dhcp

static address

masum@debian:~$ sudo ifdown enp0s3

ip addr

1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

valid\_lft forever preferred\_lft forever

inet6 ::1/128 scope host

valid\_lft forever preferred\_lft forever

2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP group default qlen 1000

link/ether 08:00:27:21:78:63 brd ff:ff:ff:ff:ff:ff

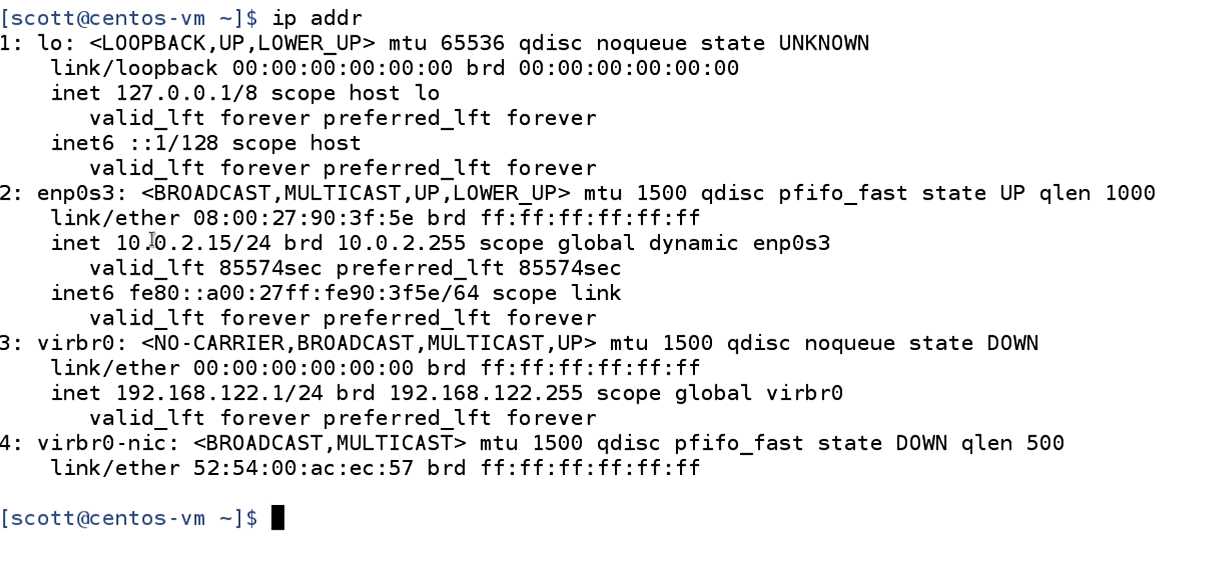
inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3

valid\_lft 85495sec preferred\_lft 85495sec

inet6 fe80::a00:27ff:fe21:7863/64 scope link

valid\_lft forever preferred\_lft forever

scrrenshot from the training video



masum@debian:~$ ip -4 addr

1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000

inet 127.0.0.1/8 scope host lo

valid\_lft forever preferred\_lft forever

2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP group default qlen 1000

inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3

valid\_lft 85236sec preferred\_lft 85236sec

permitted by applicable law.

Last login: Sun Oct 17 22:55:48 2021 from 10.0.2.2

masum@debian:~$ sudo ip link set enp0s3 down

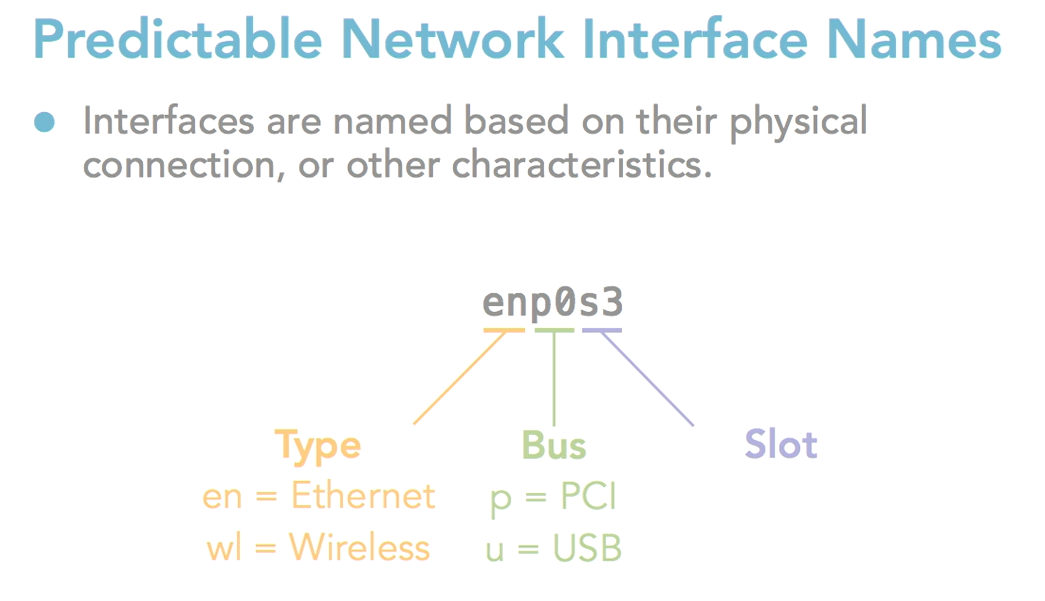
If this command is run putty on windows won’t work

masum@debian:~$ sudo ip link set enp0s3 down

masum@debian:~$ sudo ip link set enp0s3 up

ethtool enp0s3 [ this command doesn’t work on putty for debian]

enp0s3



**/etc/NetworkManager/NetworkManager.conf**

[ifupdown]

managed=false

TO:

[ifupdown]

managed=true

but as we set dhcp or static ip addresses manually that is why we set it to false

<https://superuser.com/questions/950431/how-to-install-virtual-box-guest-additions-on-debian>

<https://superuser.com/questions/776689/how-to-install-latest-virtualbox-dkms-with-latest-virtualbox-on-debian-stable-wh>

<https://community.spiceworks.com/topic/1523821-how-do-i-logon-as-root-in-debian>

<https://wiki.koha-community.org/wiki/Debian>

echo 'deb http://debian.koha-community.org/koha stable main' | tee /etc/apt/sources.list.d/koha.list

The following plugins have been configured:

rabbitmq\_stomp

Applying plugin configuration to rabbit@debian...

The following plugins have been enabled:

rabbitmq\_stomp

started 1 plugins.

Failed to enable unit: Unit /run/systemd/generator.late/koha-common.service is transient or generated.

Processing triggers for sgml-base (1.29) ...

Setting up docbook-xml (4.5-8) ...

Processing triggers for systemd (241-7~deb10u8) ...

Processing triggers for sgml-base (1.29) ...

Username : *koha\_libraryname*

sudo koha-passwd libraryname [in my case its library

Password: wr6IHxZDeZwlvg4@

Library code: IEDCRLIB

Name: IEDCR Library

https://wiki.koha-community.org/wiki/Koha\_on\_Debian#Installation\_process

Card number- 202110200001

Username: admin

Passwd : Masum01964

sudo xmlstarlet sel -t -v '/yazgfs/config/user' /etc/koha/sites/libraryname/koha-conf.xml

koha\_libraryname

Networking

sudo apt update

sudo apt install snapd

sudo snap install core

sudo snap install network-manager

setting static ip debian

masum@dspace:~$ nmcli d

DEVICE TYPE STATE CONNECTION

enp0s3 ethernet connected Wired connection 1

lo loopback unmanaged --

masum@dspace:~$ sudo ifdown enp0s3

Killed old client process

Internet Systems Consortium DHCP Client 4.4.1

Copyright 2004-2018 Internet Systems Consortium.

All rights reserved.

For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/enp0s3/08:00:27:af:55:d6

Sending on LPF/enp0s3/08:00:27:af:55:d6

Sending on Socket/fallback

DHCPRELEASE of 10.0.2.15 on enp0s3 to 10.0.2.2 port 67

masum@dspace:~$ nmcli general hostname

masum@dspace:~$ hostnamectl

Static hostname: dspace

Icon name: computer-vm

Chassis: vm

Machine ID: 22b7fd72c49d4e77aabe58307d360f82

Boot ID: 93cd34764e4d4ec8b4513236127631ea

Virtualization: oracle

Operating System: Debian GNU/Linux 10 (buster)

Kernel: Linux 4.19.0-18-amd64

Architecture: x86-64

masum@dspace:~$ sudo iptables -L

Chain INPUT (policy ACCEPT)

target prot opt source destination

Chain FORWARD (policy ACCEPT)

target prot opt source destination

Chain OUTPUT (policy ACCEPT)

target prot opt source destination

masum@dspace:~$ sudo iptables-save > rules-backup

sudo iptables-restore rules-backup

State Recv-Q Send-Q Local Address:Port Peer Address:Port

ESTAB 0 0 10.0.2.15:ssh 10.0.2.2:1192

masum@dspace:~$ ss -tln

State Recv-Q Send-Q Local Address:Port Peer Address:Port

LISTEN 0 128 0.0.0.0:22 0.0.0.0:\*

LISTEN 0 128 [::]:22 [::]:\*

masum@dspace:~$ sudo ufw delete 3

Deleting:

allow WWW

Proceed with operation (y|n)? y

Rule deleted (v6)

masum@dspace:~$ sudo status ufw

sudo: status: command not found

masum@dspace:~$ sudo ufw status

Status: active

To Action From

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

22/tcp ALLOW Anywhere

WWW ALLOW Anywhere

masum@dspace:~$ sudo apache2ctl configtest

AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message

Syntax OK

masum@dspace:~$ sudo apt install curl

masum@dspace:~$ curl 192.168.48.101

NAME UUID TYPE DEVICE

Wired connection 1 cecf0a7c-69bd-3671-871d-5441e1cc3302 ethernet enp0s3

Wired connection 2 9a207fe7-d422-3fd7-9c8c-95d7541e750c ethernet --