

# **ADVANCE-LINUX**

**Ques 1:-** What is the size of MBR and what does it contains

Ans 1:- Size of an MBR is 512 bytes & it contains the information of the first sector of the hard disk where the operating system is located and access ram. It also contains master partition table.

**Ques 2:-** In which file you can write commands which you want to run whenever Linux system starts/restarts?.

Ans 2:- .profile

**Ques 3:-** Reboot the system using runlevel.

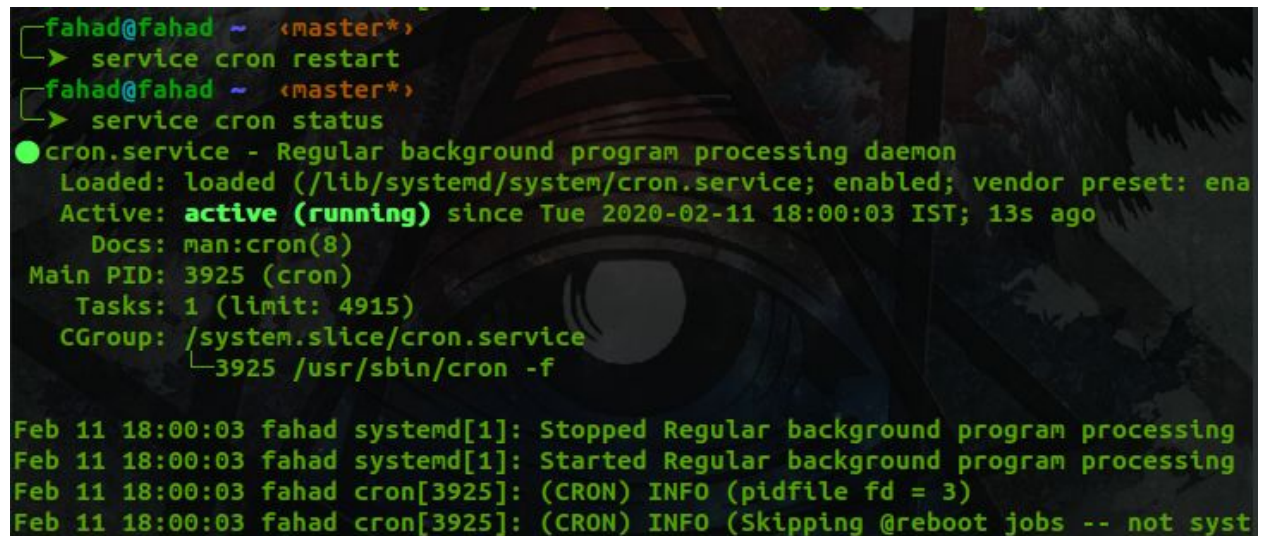
Ans 3:- **init 6** runlevel is use to restart the system

Runlevels define what tasks can be accomplished in the current state (or runlevel) of a Linux system

- 0- Halt
- 1- Single user mode (recovery)
- 2- Debian/Ubuntu default
- 3- RHEL/Fedora/SUSE text mode
- 4- free
- 5- RHEL/Fedora/SUSE graphical mode
- 6- reboot

**Ques 4:-** Restart cron service.

Ans :-

A terminal window with a dark background and a large eye graphic. The user 'fahad' is at the prompt 'fahad@fahad ~ <master\*>'. They enter 'service cron restart'. Then they enter 'service cron status'. The output shows 'cron.service - Regular background program processing daemon' is 'loaded (/lib/systemd/system/cron.service; enabled; vendor preset: enabled)' and 'Active: active (running) since Tue 2020-02-11 18:00:03 IST; 13s ago'. It also shows 'Main PID: 3925 (cron)', 'Tasks: 1 (limit: 4915)', and 'CGroup: /system.slice/cron.service'. At the bottom, there are four log messages: 'Feb 11 18:00:03 fahad systemd[1]: Stopped Regular background program processing', 'Feb 11 18:00:03 fahad systemd[1]: Started Regular background program processing', 'Feb 11 18:00:03 fahad cron[3925]: (CRON) INFO (pidfile fd = 3)', and 'Feb 11 18:00:03 fahad cron[3925]: (CRON) INFO (Skipping @reboot jobs -- not syst'.

```
fahad@fahad ~ <master*>
> service cron restart
fahad@fahad ~ <master*>
> service cron status
● cron.service - Regular background program processing daemon
   Loaded: loaded (/lib/systemd/system/cron.service; enabled; vendor preset: ena
   Active: active (running) since Tue 2020-02-11 18:00:03 IST; 13s ago
     Docs: man:cron(8)
  Main PID: 3925 (cron)
    Tasks: 1 (limit: 4915)
   CGroup: /system.slice/cron.service
           └─3925 /usr/sbin/cron -f

Feb 11 18:00:03 fahad systemd[1]: Stopped Regular background program processing
Feb 11 18:00:03 fahad systemd[1]: Started Regular background program processing
Feb 11 18:00:03 fahad cron[3925]: (CRON) INFO (pidfile fd = 3)
Feb 11 18:00:03 fahad cron[3925]: (CRON) INFO (Skipping @reboot jobs -- not syst
```

Ques 5:- Create an ext4 filesystem

Ans :- First we have to create a partition by **sudo fdisk /dev/sda**

```
fahad@fahad ~ (master*)
> sudo fdisk /dev/sda

Welcome to fdisk (util-linux 2.31.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): n
Partition type
  p   primary (1 primary, 1 extended, 2 free)
  l   logical (numbered from 5)
Select (default p): p
Partition number (3,4, default 3):
First sector (781250560-1953525167, default 781250560):
Last sector, +sectors or +size[K,M,G,T,P] (781250560-781252605, default 781252605):

Created a new partition 3 of type 'Linux' and of size 1023 KiB.

Command (m for help): n
Partition type
  p   primary (2 primary, 1 extended, 1 free)
  l   logical (numbered from 5)
Select (default p):

Using default response p.
Selected partition 4
First sector (1101561856-1953525167, default 1101561856):
Last sector, +sectors or +size[K,M,G,T,P] (1101561856-1953525167, default 1953525167): +10G

Created a new partition 4 of type 'Linux' and of size 10 GiB.
```

Now we have to list all the partition by pressing **p**

```
Command (m for help): p
Disk /dev/sda: 931.5 GiB, 1000204886016 bytes, 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
Disklabel type: dos
Disk identifier: 0x30c6e8e4

Device      Boot      Start          End      Sectors      Size Id Type
/dev/sda1                2048    781250559    781248512    372.5G 83 Linux
/dev/sda2          781252606    1101561855    320309250    152.8G  5 Extended
/dev/sda3          781250560     781252605         2046    1023K 83 Linux
/dev/sda4        1101561856    1122533375    20971520      10G 83 Linux
/dev/sda5          781252608     976562175    195309568     93.1G 83 Linux
/dev/sda6          976564224    1101561855    124997632     59.6G 82 Linux swap / Solaris

Partition 2 does not start on physical sector boundary.
Partition table entries are not in disk order.
```



Now to write and exit type **w**

To create a file system of **ext4** we have to type command **mkfs.ext4 <partition name>**

```
fahad@fahad ~ <master*>
└─> sudo mkfs.ext4 /dev/sda4
mke2fs 1.44.1 (24-Mar-2018)
Discarding device blocks: done
Creating filesystem with 2621440 4k blocks and 655360 inodes
Filesystem UUID: 33569d98-0ceb-41a6-a319-788d605cc3d7
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

**Ques 6:-** Mount the created filesystem on /partition directory.

**Ans 6:-** First we have to make a directory in **/mnt** folder by **sudo mkdir /mnt/fahad**

```
fahad@fahad ~ <master*>
└─> sudo mkdir /mnt/fahad
```

Then we have to run mount command like **mount /dev/sda4 /mnt/fahad**

We can check the mount by **mount | grep fahad**

```
fahad@fahad ~ <master*>
└─> sudo mount /dev/sda4 /mnt/fahad
fahad@fahad ~ <master*>
└─> mount | grep fahad
/dev/sda4 on /mnt/fahad type ext4 (rw,relatime,data=ordered)
```

**Ques 7.** Difference between LVM and RAID.

**Ans 7.**

Difference between RAID and LVM		
S.No.	RAID	LVM
1.	RAID is used for redundancy.	LVM is a way in which you partition the hard disk logically and it contains its own advantages.
2.	A RAID device is a physical grouping of disk devices in order to create a logical presentation of one device to an Operating System for redundancy or performance or a combination of the two.	LVM is a logical layer that that can be anipulated in order to create and, or expand a logical presentation of a disk device to an Operating System.
3.	RAID is a way to create a redundant or striped block device with redundancy using other physical block devices.	LVM usually sits on top of RAID blocks or even standard block devices to accomplish the same result as a partitioning, however it is much more flexible than partitions. You can create multiple volumes crossing multiple physical devices, remove physical devices without loosing data, resize the volumes, create snapshots, etc
4.	RAID is either a software or a hardware technique to create data storage redundancy across multiple block devices based on required RAID levels.	LVM is a software tool to manage large pool of storage devices making them appear as a single manageable pool of storage resource. LVM can be used to manage a large pool of what we call Just-a-bunch-of-Disk (JBOD) presenting them as a single logical volume and thereby create various partitions for software RAID.
5.	RAID is NOT any kind of Data backup solution. Its a solution to prevent one of the SPOFs (Single Point of Failure) i.e. DISK failure. By configuring RAID you are just providing an emergency substitute for the Primary disk. It NEVER means that you have configured DATA backup.	LVM is a disk management approach that allows us to create, extend, reduce, delete or resize the volume groups or logical volumes.

**Ques 8:-** Create a LVM

**Ans 8:-** To create a Logical Volume first we have to make a physical volume by `sudo pvcreate /dev/sda4` to see the volume `sudo pvdisplay`

```
fahad@fahad ~ <master*>
> sudo pvcreate /dev/sda4

WARNING: ext4 signature detected on /dev/sda4 at offset 1080. Wipe it? [y/n]:  WARNING: Invalid input ''.
WARNING: ext4 signature detected on /dev/sda4 at offset 1080. Wipe it? [y/n]: y
Wiping ext4 signature on /dev/sda4.
Physical volume "/dev/sda4" successfully created.
```

Then we have to create the volume group by `sudo vgcreate vol_grp1 /dev/sda4`

```
fahad@fahad ~ <master*>
> sudo vgcreate vol_grp1 /dev/sda4
Volume group "vol_grp1" successfully created
```

Then we have to create the Logical volume by `sudo lvcreate -l 20 -n logical_vol1 vol_grp1`

To see the Logical volume we have to run command `sudo lvdisplay`

```
fahad@fahad ~ <master*>
> sudo lvcreate -l 20 -n logical_vol1 vol_grp1
Logical volume "logical_vol1" created.
fahad@fahad ~ <master*>
> sudo lvdisplay
--- Logical volume ---
LV Path                /dev/vol_grp1/logical_vol1
LV Name                 logical_vol1
VG Name                 vol_grp1
LV UUID                 yl8qrB-27fA-LSWk-BBs2-yY3Y-r7eF-9bXFNZ
LV Write Access         read/write
LV Creation host, time fahad, 2020-02-11 23:58:18 +0530
LV Status                available
# open                  0
LV Size                 80.00 MiB
Current LE              20
Segments                1
Allocation               inherit
Read ahead sectors      auto
- currently set to      256
Block device            253:0
```

**Ques 9:-** Create a RAID1 device

**Ans 9:-**

```
fahad@fahad ~ (master*)  
➤ sudo mdadm --create --verbose /dev/md0 --level=1 --raid-devices=2 /dev/sda3  
/dev/sda4  
mdadm: Note: this array has metadata at the start and  
may not be suitable as a boot device. If you plan to  
store '/boot' on this device please ensure that  
your boot-loader understands md/v1.x metadata, or use  
--metadata=0.90  
mdadm: size set to 10484736K  
mdadm: largest drive (/dev/sda3) exceeds size (10484736K) by more than 1%  
Continue creating array? y  
mdadm: Defaulting to version 1.2 metadata  
  
mdadm: array /dev/md0 started.  
fahad@fahad ~ (master*)  
➤  
fahad@fahad ~ (master*)  
➤ cat /proc/mdstat  
Personalities : [raid1]  
md0 : active raid1 sda4[1] sda3[0]  
10484736 blocks super 1.2 [2/2] [UU]  
[==>.....] resync = 14.7% (1543552/10484736) finish=3.1min sp  
eed=46774K/sec  
  
unused devices: <none>
```

**To check raid devices types**

➔ #mdadm -E /dev/sda3 /dev/sda4  
#mdadm --detail /dev/md0



```

fahad@fahad ~/Desktop (master*)
└─> sudo mdadm -E /dev/sda3 /dev/sda4
[sudo] password for fahad:
T34Sorry, try again.
[sudo] password for fahad:

/dev/sda3:
    Magic : a92b4efc
    Version : 1.2
    Feature Map : 0x0
    Array UUID : 80bde1f8:f92a3050:06aedd8:148fc7f2
    Name : fahad:0 (local to host fahad)
    Creation Time : Wed Feb 12 15:38:04 2020
    Raid Level : raid1
    Raid Devices : 2

Avail Dev Size : 18446744073709551614
    Array Size : 10484736 (10.00 GiB 10.74 GB)
    Used Dev Size : 20969472 (10.00 GiB 10.74 GB)
    Data Offset : 2048 sectors
    Super Offset : 8 sectors
    Unused Space : before=1896 sectors, after=18446744073688582142 sectors
    State : clean
    Device UUID : c64ed27b:55eba42c:cf356f96:29d5b698

    Update Time : Wed Feb 12 15:41:25 2020
    Bad Block Log : 512 entries available at offset 136 sectors
    Checksum : b4e8f467 - correct
    Events : 17

    Device Role : Active device 0
    Array State : AA ('A' == active, '.' == missing, 'R' == replacing)
/dev/sda4:
    Magic : a92b4efc
    Version : 1.2
    Feature Map : 0x0

```

**Ques 10:-** Create a swapfile of 500Mb

**Ans 10:-**

```

fahad@fahad ~ (master*)
└─> sudo fallocate -l 500M /swapfile
fahad@fahad ~ (master*)
└─> sudo mkswap /swapfile
mkswap: /swapfile: insecure permissions 0644, 0600 suggested.
Setting up swapspace version 1, size = 500 MiB (524283904 bytes)
no label, UUID=d220861f-f750-48ba-a1c2-c082c01a6ddd
fahad@fahad ~ (master*)
└─> sudo chmod 600 /swapfile
fahad@fahad ~ (master*)
└─> sudo swapon /swapfile
fahad@fahad ~ (master*)
└─> sudo swapon -s

```

Filename	Type	Size	Used	Priority
/dev/sda6	partition	62498812	0	-2
/swapfile	file	511996	0	-3

**Ques 11:-** Set setuid and setgid on two different file

**Ans 11:-** To **setuid** run command **sudo chmod u+s fahad.txt**

```

fahad@fahad ~ (master*)
[> ls -ld fahad.txt
-rw-r--r-- 1 fahad fahad 0 Feb 12 00:11 fahad.txt
fahad@fahad ~ (master*)
[> sudo chmod u+s fahad.txt

fahad@fahad ~ (master*)
[>
fahad@fahad ~ (master*)
[> ls -ld fahad.txt
-rwSr--r-- 1 fahad fahad 0 Feb 12 00:11 fahad.txt

```

To **setgid** run command **sudo chmod g+s fahad1.txt**

```

fahad@fahad ~ (master*)
[> ls -ld fahad1.txt
-rw-r--r-- 1 fahad fahad 0 Feb 12 00:11 fahad1.txt
fahad@fahad ~ (master*)
[> sudo chmod g+s fahad1.txt
fahad@fahad ~ (master*)
[> ls -ld fahad1.txt
-rw-r-Sr-- 1 fahad fahad 0 Feb 12 00:11 fahad1.txt

```

**Ques 12:-**What is the use of Sticky bit.

**Ans 12:-**Sticky bit is used on directories where each user have write permissions but only on their files, not on others.

An example of sticky bit is /tmp where each user can write its data but cannot write other user's files.

```

fahad@fahad ~ (master*)
[> ls -ld fahad1.txt
-rw-r--r-- 1 fahad fahad 0 Feb 12 00:11 fahad1.txt
fahad@fahad ~ (master*)
[> sudo chmod o+t fahad1.txt
fahad@fahad ~ (master*)
[> ls -ld fahad1.txt
-rw-r--r-T 1 fahad fahad 0 Feb 12 00:11 fahad1.txt

```

**Ques 13:-** Create a user and add it to one secondary group

**Ans 13:-** first we have to create a user with **sudo adduser newuser**

```

fahad@fahad ~ (master*)
└─> sudo adduser newuser
[sudo] password for fahad:
Adding user `newuser' ...
Adding new group `newuser' (1002) ...
Adding new user `newuser' (1002) with group `newuser' ...
Creating home directory `/home/newuser' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for newuser
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
fahad@fahad ~ (master*)
└─> id newuser
uid=1002(newuser) gid=1002(newuser) groups=1002(newuser)

```

Here we can see that the user is added to newuser group

By **usermod -G <secondary group> <username>** it will add the secondary group to the user

```

fahad@fahad ~ (master*)
└─> sudo usermod -G fahad newuser
fahad@fahad ~ (master*)
└─> id newuser
uid=1002(newuser) gid=1002(newuser) groups=1002(newuser),1000(fahad)
fahad@fahad ~ (master*)
└─>

```

**Ques 14:-** Lock this user.

**Ans 14:-** **sudo usermod -L newsuser** now if we have to check the status of the user we type **sudo passwd --status newuser**

```

fahad@fahad ~ (master*)
└─> sudo usermod -L newuser
fahad@fahad ~ (master*)
└─> sudo passwd --status newuser
newuser L 02/11/2020 0 99999 7 -1

```

**Ques 15:-** Give this user full access (without password).

**Ans 15:-** By specifying the user in to the **sudoers** file we and setting all permission to **ALL** we can give full access to user



```
Defaults      secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:$
# Host alias specification
# User alias specification
# Cmnd alias specification
# User privilege specification
root    ALL=(ALL:ALL) ALL
newuser ALL=(ALL:ALL) ALL
# Members of the admin group may gain root privileges
%admin   ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:
```

**Ques 16:-**Delete the create user after taking backup of it home directory.

**Ans 16:-** By using `deluser --remove-home --backup-to /tmp/userbackup newuser` its backup is being stored in the `/tmp/userbackup/newuser.tar` file.

```
fahad@fahad ~ <master*>
└─> sudo deluser --remove-home --backup-to /tmp/userbackup newuser
Looking for files to backup/remove ...
Backing up files to be removed to /tmp/userbackup ...
backup_name = /tmp/userbackup/newuser.tar
/bin/tar: Removing leading '/' from member names
Removing files ...
Removing user 'newuser' ...
Warning: group 'newuser' has no more members.
Done.
fahad@fahad ~ <master*>
└─> 
```

**Ques 17:-** Create a file with some content. Change all lower case letter to upper case letter and save output to another file using redirections.

**Ans 17:-** **tr** command is use to convert the text

```
fahad@fahad ~/Desktop <master*>
[> cat lw.txt
this is going to be convetted into upper case
fahad@fahad ~/Desktop <master*>
[> tr '[a-z]' '[A-Z]' < lw.txt | tee > output.txt
fahad@fahad ~/Desktop <master*>
[> cat output.txt
THIS IS GOING TO BE CONVETTED INTO UPPER CASE
fahad@fahad ~/Desktop <master*>
[> █
```

**Ques 18:-** Set nice value of a process to -1.

**Ans :-** first we have to list the processes with the **top** command

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2986	fahad	20	0	2925832	335388	147884	S	43.9	2.1	3:16.55	Web Content
2227	fahad	20	0	3876804	473564	169932	S	11.9	2.9	21:31.81	firefox

Run command

**Renice -n -1 -p 2227 -u fahad** it will change the nice value to -1

```
fahad@fahad ~/Desktop <master*>
[> top | grep 2227
2227 fahad      19  -1 3878636 485328 170944 S  11.1  3.0 22:07.98 firefox
```

Ques 19:-Get list of all files used by "telnet".

Ques 20:- Check if port 22 is listening using netstat and telnet command.

```
fahad@fahad ~/Desktop <master*>
> dpkg-query --listfiles telnet
/.
/usr
/usr/bin
/usr/bin/telnet.netkit
/usr/share
/usr/share/doc
/usr/share/doc/telnet
/usr/share/doc/telnet/BUGS
/usr/share/doc/telnet/README.gz
/usr/share/doc/telnet/README.telnet
/usr/share/doc/telnet/README.telnet.old.gz
/usr/share/doc/telnet/changelog.Debian.gz
/usr/share/doc/telnet/copyright
/usr/share/lintian
/usr/share/lintian/overrides
/usr/share/lintian/overrides/telnet
/usr/share/man
/usr/share/man/man1
/usr/share/man/man1/telnet.netkit.1.gz
/usr/share/menu
/usr/share/menu/telnet
```

Ans 20:-<USING TELNET> telnet <host IP> <port no.> if the port is down it won't allow to connect

```
fahad@fahad ~/Desktop <master*>
> telnet 192.168.0.106 22
Trying 192.168.0.106...
Connected to 192.168.0.106.
Escape character is '^]'.
SSH-2.0-OpenSSH_7.6p1 Ubuntu-4ubuntu0.3
```

<USING NETSTAT> sudo netstat -plnt | grep 22 it tells whether the port is listening or not

```
fahad@fahad ~/Desktop <master*>
> sudo netstat -plnt | grep 22
tcp        0      0 0.0.0.0:22          0.0.0.0:*          LISTEN      1070/sshd
tcp6       0      0 :::22             :::*               LISTEN      1070/sshd
```

Ques 21:- Create a cron job which runs once in a week at 23:45.



Ans 21:- `crontab -u fahad -e` for creating the crontab

```
fahad@fahad ~/Desktop (master*)
> crontab -u fahad -e
crontab: installing new crontab
fahad@fahad ~/Desktop (master*)
> crontab -u fahad -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
45 23 * * 0 /bin/bash /home/fahad/flr.sh
```

Ques 22:- Difference between dig and traceroute

Ans 22:- **Dig** is a DNS lookup utility. It performs DNS lookups and displays the answers that are returned from the name servers that were queried.

```
fahad@fahad ~ (master*)
> dig www.gogle.com

;<<<>> DiG 9.11.3-1ubuntu1.11-Ubuntu <<<>> www.gogle.com
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 40596
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;www.gogle.com.                IN      A

;; ANSWER SECTION:
www.gogle.com.                299     IN      A      172.217.167.227

;; Query time: 93 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Wed Feb 12 00:22:30 IST 2020
;; MSG SIZE rcvd: 58

fahad@fahad ~ (master*)
> █
```

**Traceroute** tracks the route packets taken from an IP network on their way to a given host. It utilizes the IP protocol's time to live (TTL) field and attempts to elicit an ICMP TIME\_EXCEEDED response from each gateway along the path to the host.

```
allowed value
fahad@fahad ~ (master*)
> traceroute www.google.com
traceroute to www.google.com (172.217.166.196), 30 hops max, 60 byte packets
 1 * * *
 2 * * *
 3 * * *
 4 72.14.208.49 (72.14.208.49)  8.051 ms  8.048 ms  9.570 ms
 5 * * *
 6 172.253.67.98 (172.253.67.98)  9.757 ms 64.233.175.100 (64.233.175.100)  7.788 ms 209.85.252.44 (209.85.252.44)  5.076 ms
 7 66.249.95.75 (66.249.95.75)  6.187 ms 74.125.243.98 (74.125.243.98)  7.009 ms 66.249.95.75 (66.249.95.75)  4.486 ms
 8 del03s13-in-f4.1e100.net (172.217.166.196)  4.139 ms  4.095 ms 108.170.251.113 (108.170.251.113)  7.844 ms
fahad@fahad ~ (master*)
>
```