Osi layers:

1. Application layer: this layer is used to generate data, interact with the end user to provide services.
   Devices: Http/https/browser/client.
2. Presentation layer: this layer is responsible for data encryption and data compression for efficient transfer of data.
   Tools: SSL, TLS(data encryption), jpeg,gif.
3. Session layer: this layer is responsible for maintaining and creating sessions between client and server.
   Tools: RPC, NetBios
4. Transport layer: this layer is responsible for reliable data transfer, here we select which protocol we want to use, to transfer data between client and server.
   Tools: TCP,UDP
5. Network layer: this layer is responsible for data transfer between different devices such as router, and manages data routing.
   Tools: IP,router, ICMP(internet control message protocol), NAT, router,firewall
6. Data link layer: this layer is responsible to transfer data inside a local network, using the MAC address of the machine.
   Tools: ARP, MAC address, wifi, switch,hub
7. Physical layer: the lan cable that is attached to our computer for which we transfer our data to the network.

ICMP is used to send error messages that we encounter in the network layer, this is used by commands such as ping and traceroute to give user feedback about the network status.

MAC address is not necessarily attached to NIC card, in devices with no ethernet port it is connected to wifi card.
TCP protocol is connection oriented not stateful, while UDP is not connection oriented protocol

Network terms:

1. Echo : used to display text on the terminal if we use Echo ‘Hello world’ in that case it will print hello world on the screen.
2. Ping: it uses ICMP protocol basically used to check if some server is accessible or not.
3. traceroute : it is used to trace the route of a packet it takes from your device to servers destination, it also use traceroute

4. ARP: address resolution protocol, used to point the address to a specific device, that is we can convert an IP address to MAC address using this, used in the data link layer. ARP uses broadcasting, to find the receiving device MAC address, it will broadcast the receiving packet, the destination server will respond.

TCP/IP layer:

1. Application layer: user A composes an email and clicks send the email client breakdown the email into smaller packets each packet includes:
    1. Source IP address
    2. Destination ID address
    3. Other relevant information(payload, header)
2. Transport layer: TCP segments the data into smaller units for reliable transmission it adds sequence number and checksum to ensure data integrity and order
3. Network layer: it adds its own header to each tcp segment, including:
    1. Source IP address
    2. Destination ip address
    3. Time to live, this creates an IP datagram.

4. Data link layer:

Security Groups:
Subnet means dividing a large group of devices into multiple devices, such that we can define some common rules for our group.

What is a subnet mask?
An ip address: 192.168.1.0/24, now 24 represents the subnet mask, meaning starting 24 bits will remain the same within the ip address.

How to determine the number of available IP addresses in a subnet ?
IP address: 192.16.1.0/24, then number of available IP address will be, 2^32 - 2^24

Nat(network address translation):
This is used to direct the packet in the local network, inside a private subnet. Where it replaces the private ip address to public address.

How do we identify where to pass the packet, as the incoming packet only has the gateway address ?

So basically whenever a packet is being sent from the machine to the outside world we create an entry in NAT table, where we write the source IP address(machine's address which requested it), destination IP(address of the gateway of what we requested), and port number to identify where the service is running.

HTTPS uses SSL internally to send the data in secure form from 1 place to another.

SSL(secure socket layer): this helps us establish secure connection between host and server.

Asymmetric encryption: in this we use 2 different keys to encrypt our data that is automatically handled by https. We use public key to encrypt while private key to decrypt.

Symmetric encryption: in this we use the same key for encryption and decryption.

Whenever we make a request to the end server, first we need to verify the certificates issued by certificate authority, now there might be an issue when someone gets access to our packets while verifying the security certificate, to avoid this situation we store all the valid certificates to our own browser.

API Gateway: it is used in distributed systems, to route the request based on paths to different microservices. We can add services such as authentication/authorization in the api gateway.

IPSEC(internet protocol security) is very similar to firewalls, but in addition to firewalls that only provide functionality of traffic filters, here IPSEC can also provide features such as authentication.

Microservices:

Async communication follows pub/sub, queue, for pub sub think about sns,kafka, queue, think sidekiq in async communication we don't have to use load balancer.

Routers vs switches
Routers are used to connect 2 or more networks together in a public network
Switches are used to connect multiple devices together in a private network
Routers use NAT to make requests

Switches use MAC address

Stack VS heap

These are two types of memory available to use stack and heap. The issue with stack is it has limited storage but it is faster, so while creating instances of class, we use heaps because they need a large amount of memory. We have a call stack associated with each process, now whatever process wants to declare such as primitive variables and function calls will be saved in the call stack for faster access, so when we declare an instance of a class in the heap memory, the object stays in heap memory but a pointer is stored in the call stack for fast access.

IP address

1. Public IP address: this is the ip address which is unique and used to make requests over the network, provided by ISP, can change when we connect to a new network.
2. Private IP address: this is the IP address which we use in our private network.
3. Elastic IP: this is provided by cloud providers such as AWS such that we always get the same Public IP for a device whenever it is connected to the internet.

API gateway VS router

Both are used to transmit packets, but the difference is API gateway provides more features such as IP conversion from IPV6 to IPV4 and vice versa, on top of services such as NAT.

But the router dont provide such features.

GIT/version controls:

SCM stands for source code management done on my local machine.

Vcs stands for version control like git hub.

Disadvantages of scm:

1. Manual management
2. Back up
3. Collaboration

Advantages of SCM:

1. Secure
2. fast

Central repository: a single main repository is managed, just like github due to this we can easily collaborate.

Distributed repository:
Means repo is managed at both the places, local and remote. Just like in github where we have a .git folder now most of the operations are done locally such as maintaining versions that's why git is fast.

Fast forward merge:

1. Git fast forward merge is faster than normal merge.
2. We don't need to resolve conflicts in this case
3. We don't need to creating a different commit to show merge between branches
4. During git fast forward commit, the position of the current master branch does not change. That's why, when we merge we just change the head pointer to feature branch head.

Git reset hard/soft/hybrid:

1. Git hard: in hard reset we unstage and match the current directory with the commit.
2. Git soft: in soft reset we move the head to the mentioned commit, but we keep the later changes staged.
3. Git hybrid: in hybrid reset we move the head to the mentioned commit, while we maintain properties of soft and hard that changes persist in the current working directory but not staged.

How git works internally:

1. Git maintains 2 copies of our codebase, one on our local inside .git folder, another on remote such as github.
2. Git divides our code in 4 areas, working directory, staging area, local repo, remote repo.
3. When we add our changes, they are moved to the staging area, when we commit our changes they are moved to local repo, when we push our changes they are moved to remote repo.

How to restore last changes:

1. Git reset defined above.
2. Git revert this command creates a new commit while reverting the changes.

Creating remote/ local repo:

1. We can create local repo using git init. This will initialize the local repo.
2. After that we create a remote repo, than we set the origin of local repo
3. If we create a new branch and use command git push, then a pull request will be created
4. If we directly merge our changes in the master branch then changes directly get pushed without creating pull requests.

Databases:

atomicity : a group of sql instruction that should either complete together or fail together

Consistency: a data which should remain consistent throughout the transactions

Isolation: two transactions are isolated means they are working independently from each other, two transactions can maintain different types of isolations.

Dirty read: when two transactions are working together on the same data, one of the transactions reads the data that was updated by 2 transactions, which was not completed till then.

Durability: means data can be reverted to previous form if the system crashes in between a transaction. This can be implemented using a write ahead log.

Normalization:

First normal form: each of the column needs to be atomic, like if we create a column remark and starts to store every misc information in it, then it violates first normal form

Second normal form:

Third normal form: all the columns should depend on the primary key, not on the non primary key.

LINUX:

Linux features:

1. **open sourced:** linux is open source meaning anybody can contribute to the development of linux.
2. **Portable:** we can use linux on a wide range of hardwares. Such as iot
3. **Security:** we can store data in encrypted form in harddisk inside a linux
4. **Multi-user:** multiple users can use linux at once, meaning multiple users can have different effects on linux machines at the same time.

Linux works on 4 different layers:

- **Application layer:** this layer is used to run shell scripts, or applications such as web browsers
- **kernel :** kernel is used to interact with the hardware, kernel is always executed in the protected region of ram, such that applications can’t crash the system, kernel allocates memory to applications to run, and provides them access to hardware.
- **Shell:** using shell we can interact with the linux kernel, here the user can send the command to the kernel, after execution the kernel sends the response to the shell.
- **Hardware**: this is the hardware part of our system, such as harddisk, ssd, ram, rom.

Swap space: this is space which is being used for swapping the programs from the primary memory to secondary memory.

What happens when a operating system starts:

BIOS: Bios contains our hardware configuration, it contains the details of how to start our pc. It is stored in ROM(read only memory), but nowadays it is stored in a flash drive such that it can be updated programmatically.

MBR: master boot record, it provides us the option to choose which operating system we want to load in case of multi operating system.

Boot loader: contains the configuration about how to load the operating system into the memory.

Kernel: kernel is used to interact with the hardware, in case of I/O call we need to switch the context form user to kernel mode.

INIT: init is a process which runs the startup apps, loading env variables to the memory.

Features of operating system:

1. Allocates the memory to the program which is currently running.
2. Helps in switching the context of a process, from kernel to user.
3. Process management: allocate deallocate the memory, stack heap these all are managed by OS

4. Device management: mouse, keyword

Root directory: this is the directory, from where every file can be accessed in the linux, that is root is the parent directory in linux represented by “/”

- CD “./” means inside the current directory.
- CD “/” means inside the root directory.
- CD “../” means inside the parent directory.

ETC folder: is used to store all the configuration related to pc such as ip address of dns, logs of pc, device connected.

In linux everything is a file, so when we do ls -ltr in linux, we see different types of files available in linux. These files are denoted by chars of various types such as d for normal directory, s for symbolic file.

SSH connections:

- SSH(secure socket shell): this is used to form a secure connection to a remote device using our local terminal, using this we can access the remote device terminal on our own device.
- From our local device we can connect to a remote device using a SSH terminal, now on our local device contains the private key and public key, while the remote device stores the public key.
- When the connection is established for the first time, our SSH client send the request to server, then our server sends an encrypted message back to the client, which we decrypt using the private key, and sends the decrypted message back to the server, if the message matches then connection request is accepted.

How a linux system is started?

- **POST (power on self test):** this checks if all the hardwares are working correctly or not
- **Boot loader:** Bios load the boot loader to the memory, after this we identify where our kernel images are stored and it is loaded into the memory.
- **Kernel Init**: kernel is loaded in memory, it starts the hardwares present in the system for functioning
- **Init process:** this process is the first process that runs when a system becomes active, this is used to run the applications which run whenever the system starts, such as date and time.

Linux Commands:

1. **Echo $path:** this is used to print the path that we have set up in our machine.

2. **Ls -ltr:** ls is used to list all directories, LTR means long list format, t time of updation, r means reverse.
3. **ls -ltr shell.txt -rwxr-xr-x 1 USTR+291057 4096 31 Jan 28 12:01 shell.txt***

4. **Echo “text” >> file_name:** this command is used to create a file, >> means we are appending the content to the file, > meaning we want to create a new file with that content, else it will replace the pre-existing content of the file.

5. **Chmod:** this command is used to change the file permission that a user has over a file. We can use tags such as r-read, w-write, x-execute.
    - We provide permission to 3 users to a file, we use 3 bits to represent this, 1st bit is used to give permission to the owner, the 2nd bit is used to give permission to group, 3rd bit is used to give permission to all other users. We can create a group to provide permission to users in a group.

6. **Sudo su:** this is the command, using which we can change the user, and convert this to super user.

7. Helping commands:
    - **MAN**(manual page): most descriptive page that is available to read, provides example usage of commands with various tag we can use. **MAN CD**
    - **INFO**(online page): it opens the online documentation of the commands, **INFO CD**
    - **whatis**(one line description): **whatis CD**
    - –**help**: provide a brief built-in summary of the command usually stored locally. **Cd –help**

8. Commands to clear the terminal:
    1. Reset: it clears the terminal and resets the colour, other settings of the terminal also
    2. Clear: it clears the terminal but keeps other settings intact.

9. **LS** command is used to list all the files inside the directory:

- LS -a: prints all the files inside the directory including hidden
- LS -l: prints files in long format
- LS -h: prints file in human readable form

10. **Who command:** is used to get the details of current logged in user

11. **Wildcards: *, ?, [], ^, -**
    - * is used to match file names, it works well with ls ***.txt**
    - **?** is used to match a single char, **ls f?.txt**
    - [] is used to pass an array of characters, we list all the files which match with that array of files. **LS f[as].txt or LS f[a-s].txt**

12. **Mkdir**: make directory, **rmdir**: remove empty directory, **rmdir -rf**: to remove non empty directory

Difference between relative and absolute path ?
Relative path starts from the current working directory while absolute path starts always from the root directory.

Some commands to be used in the VI:

1. i(insert): this is used when we want to start typing in the vi
2. a(append): when we want to append a char, at the current cursor position.
3. I(insert): this is used to start typing at the start of the line
4. A(append): this is used to append a character at the end of the current line.
5. o(enter a new line): creates a new line below the current position of cursor
6. O(enter a new line): create a new line above the current position of the cursor.
7. q:(to exit) this is used to exit if no change is done
8. q!(to exit) this is used to exit the file without saving
9. wq(to save and exit)
10. q!: exit without saving.
11. X: to delete the char
12. DD: to delete the line
13. yy: used to copy current line
14. p: used to paste the copied content.
15. U: is used to undo the recent changes
16. Ctrl+r: to redo the changes

Some generally used commands:

1. Touch: used to create a new file empty files
2. Echo “text” >> file_name: this is used to create/update a file with text
3. Echo “text” > file_name: this is used to overwrite the existing content of the file. Arrow is important.
4. cat: stands for concatenate, this can be used to print the content of the file on the terminal, this can also be used to create a file. **Cat “text” >> file_name**
5. cp: this is used to copy the content of one file to another. cp file1 >>file 2.
6. rm is used to delete the file, **rm file.txt**
7. Find command: used to find a file, can be used with different flags, for specific search.

Process related commands:

1. Top: used to give details of process consuming maximum resources cpu/memory.
2. Kill p_id: used to kill the process gracefully
3. kill -9 p_id: used to kill the process forcefully
4. df: this is used to show the disk usage, -h flag is used to show the data in human readable form.
5. Netstat: display network connection/ statistics
6. Curl: is used to get data from the internet.
7. ps aux: this is used to display the detailed view of currently running process

Command for pattern matching in linux:

1. Awk: this is used to filter the data based on some columns, we can use this to filter the data, this is more memory efficient than grep, cause it sequentially loads data to data on memory, we use $ to select which column we need. **awk '/pattern/ {print $2}' file.txt**
2. Grep is used to search data in 1 go, this loads whole data to memory at once, then starts searching.
3. Cut: