1. Good Evening
2. Lecture begins at 9:08pm
3. Topic → Computer Networks

Course Structure

1. Introduction
2. Layering Architecture — 1
3. HTTP, DNS
4. Cookies
5. Client-Server vs P2P — 2

Theory

6. TCP
7. UDP — 3

"

8. Socket Programming
9. Sockets vs Ports — 4

Practical

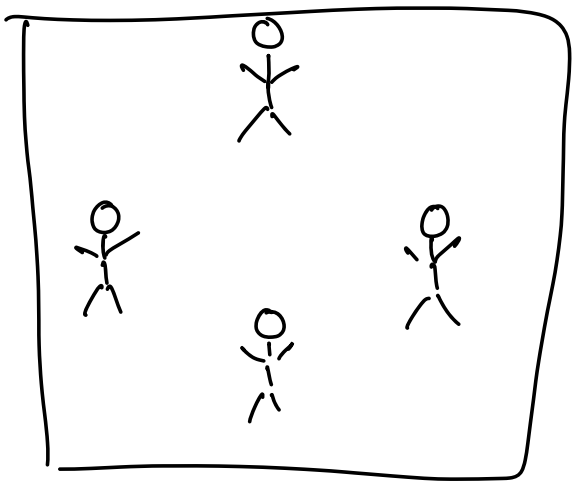
Agenda

1. Course Structure ✓
2. Introduction to Computer Networks
3. Protocols & RFC
4. Network addressing
5. Network Architecture
   - OSI Model (Theoretical)
   - TCP/IP Model (Practical)

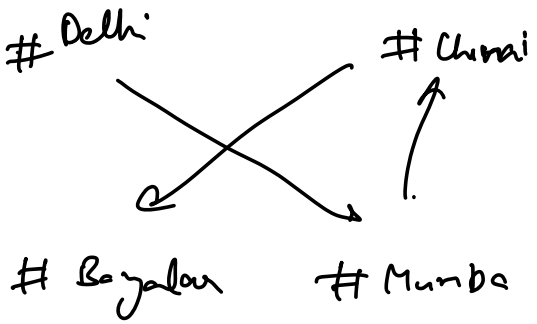
Computer Network

Examples of a network →

* A social network



* Railway Network



→ A group of connected objects

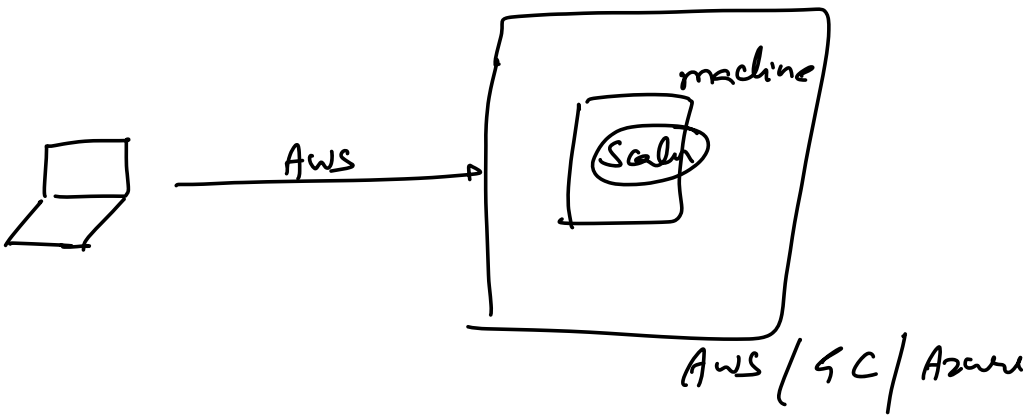
Computer Network → A group of connected computers.

Internet → A global network of computers

Advantages of a Computer Network

1. Communication

2. Sharing Resources



Resources of a machine on AWS (cloud)

- CPU
- Disk
- RAM

Historical Beginning

→ US Army

ARPA

ARPANet

Protocols

Grammer

☆ Language → Standard set of rules/ instructions

[Computer also need rules to interact with each other

☆ Protocal → A set of rules for transmission of data btw two devices.

examples.
- → http, ftp
- → TCP, UDP
- → IP

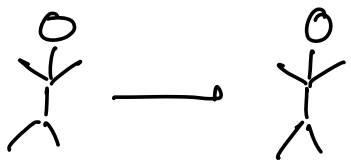
☆ A protocal is a set of rules
☆ Of needs to be documented
☆ Some organization should keep a repository of documents.

IETF [Internet Engineering Task Force]

Registering a new protocal

1. IETF website
2. Documents with rules & RFC {Request for comments}
3. Approved.

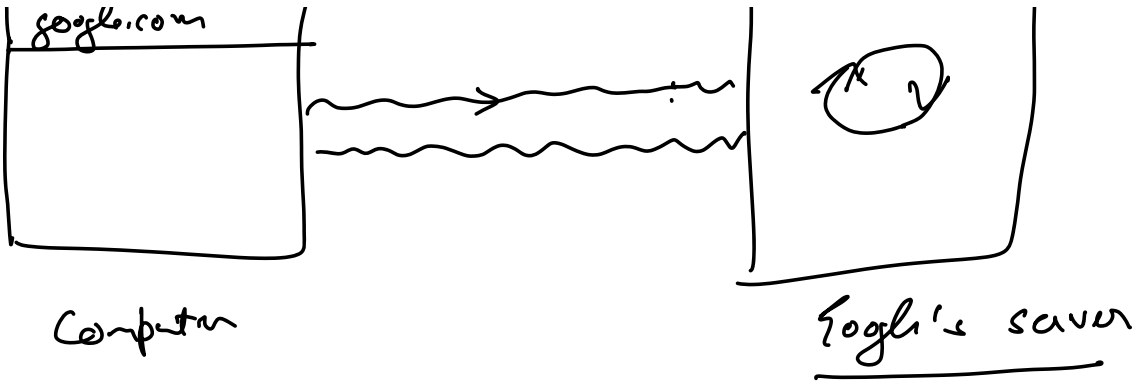
Addressing



☆ 9 don't have to address

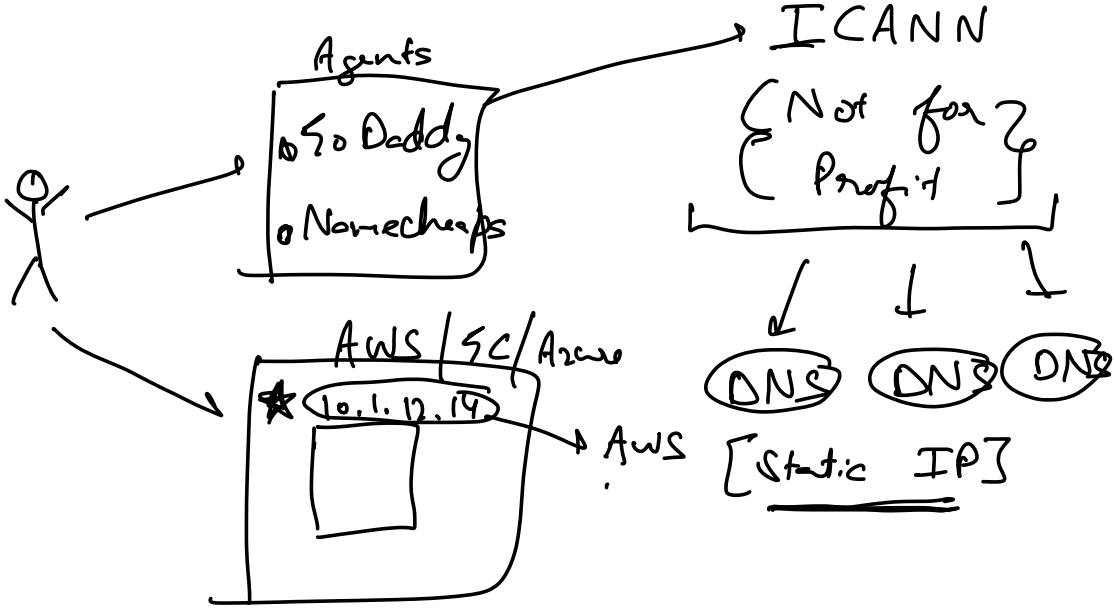
☆ IP address

Computer gets address from ISP to talk to each other



→ We don't know google's server's IP address

→ Our machine knows IP address of Google server.



1. Godaddy / NC & check if scaler.com is available
2. AWS / GC / Azure & buy a machine
3. Map domain name with

3. Godaddy | NC — IP address

scaler.com (DN) → 10.1.12.14 (Aws)

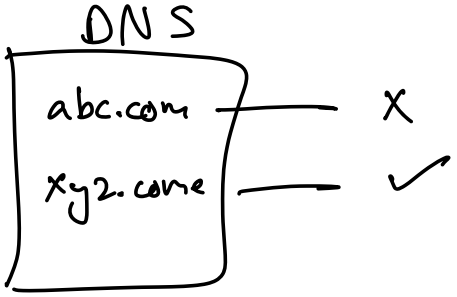
4. ICANN

5. DNS : Domain Name Servers full information from ICANN.

DNs are generally maintained by

ISP — Vodafone, Airtel, Jio, Google ✡

✡ Blocked website : Proxy



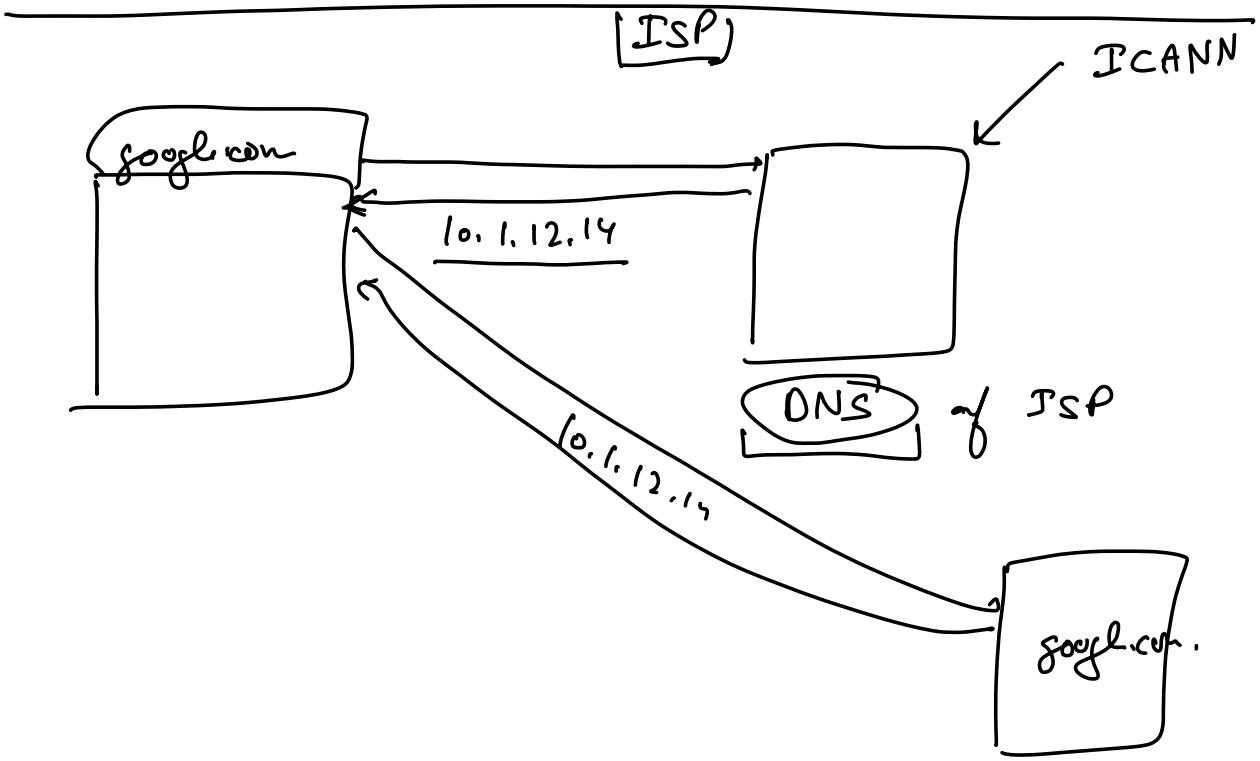
✡ 1. Azure By → IP ✡

2. Godaddy → update

3. Run ~~a~~ server on Azure

0 - 255 . — . — . —

$2^{32} \sim 10^9$ 7×10^9

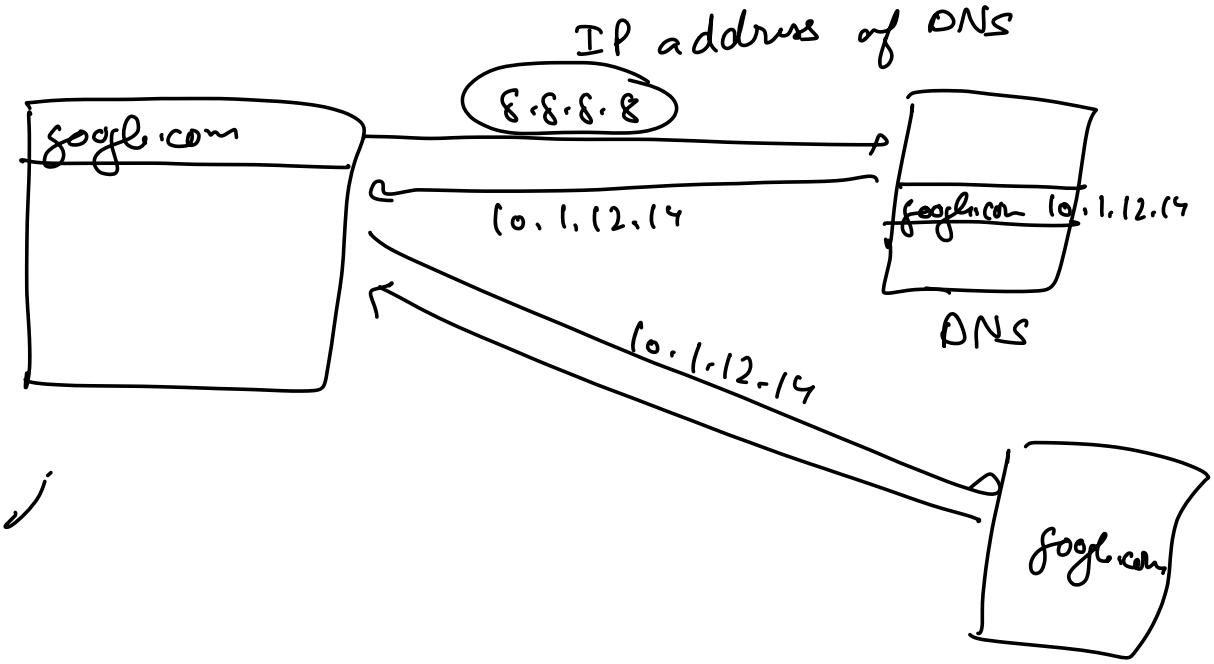


★ How is our machine able to connect to DNS?

List of DNS is stored in our

settings → ISP

8.8.8.8 → DNS Google



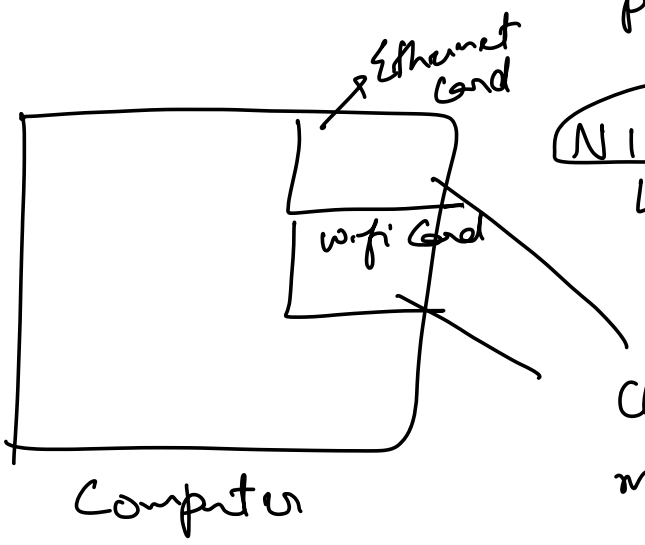
✡ DNS IP address are stored on client machines who are connected to the network.

Physical Addresses or MAC Address

↓

Media Access Controller

✡ Wireless Adapter → makes the machine wi



powered

NIC / NIA

↳ Network Interface Card / Adapter

Chips that allow machines to connect to networks

IEEE → Range of addresses for NIC's & gives the ranges to different hardware creators.

Apple → A range of NIC addresses from IEEE

:

HP

:

:

✱ Can a machine has two MAC addresses?

↳ 1 per NIC

MAC address → 1000 — 1000

↓

Apple by IEEC

IP address	MAC Address
→ Computer get's IP address via ISP.	→ Hardware Based permanent address
→ Can change	→ Can't change
→ Given by ISP	→ Given by hardware creator.

---

48 bits → $2^{48} = 10^{15}$

$\frac{1000000}{7} \times 10^9$

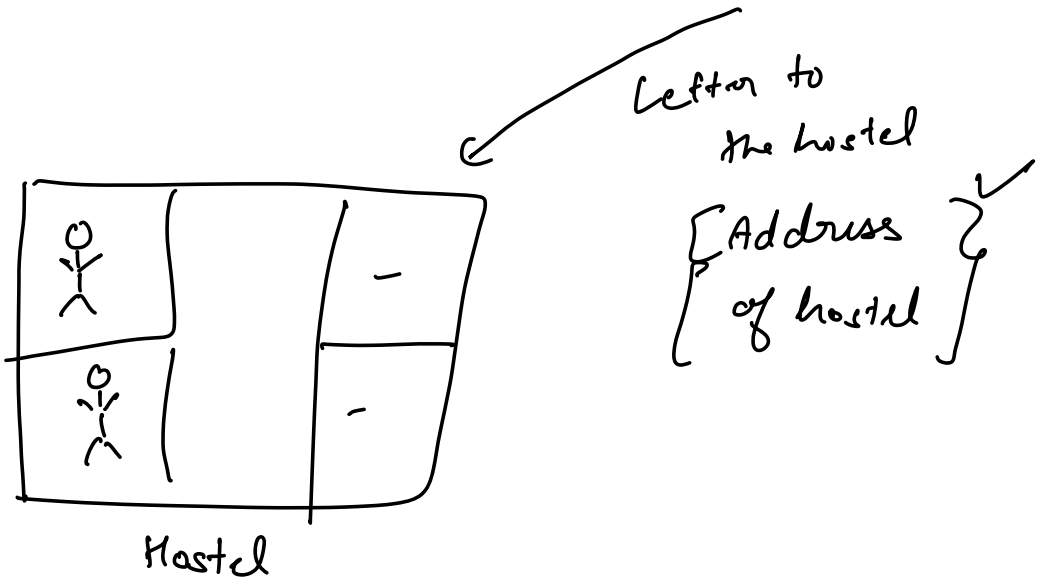
7×10^9

Break → 10:34 → 10:44

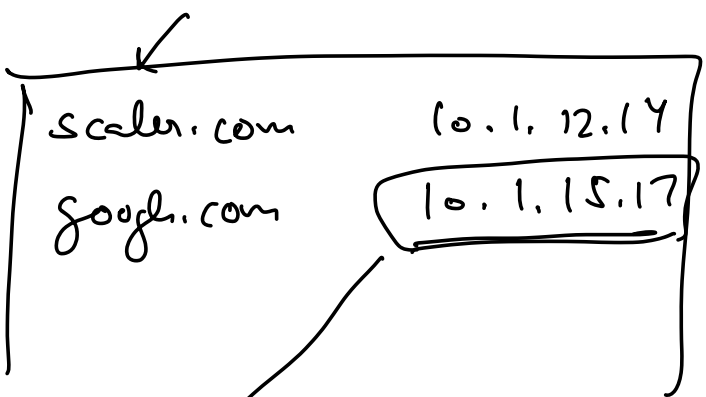
* Ports

* Layering
  - OSI Model → Theoretical
  - TCP/IP Model → Practical

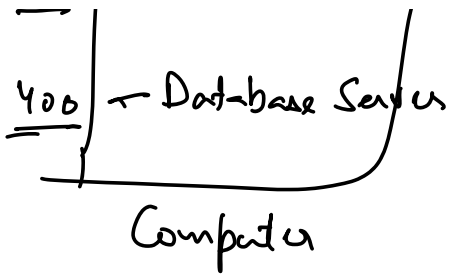
Ports



IP addresses : DNS



So ← Web Server



10. 1. 12. 14 ⟶ not enough

10. 1. 12. 14 : 80 → Application on the machine

10. 1. 12. 14 — Machine

80 — port number

IP address ⟶ machine

Port ⟶ Application on a machine.

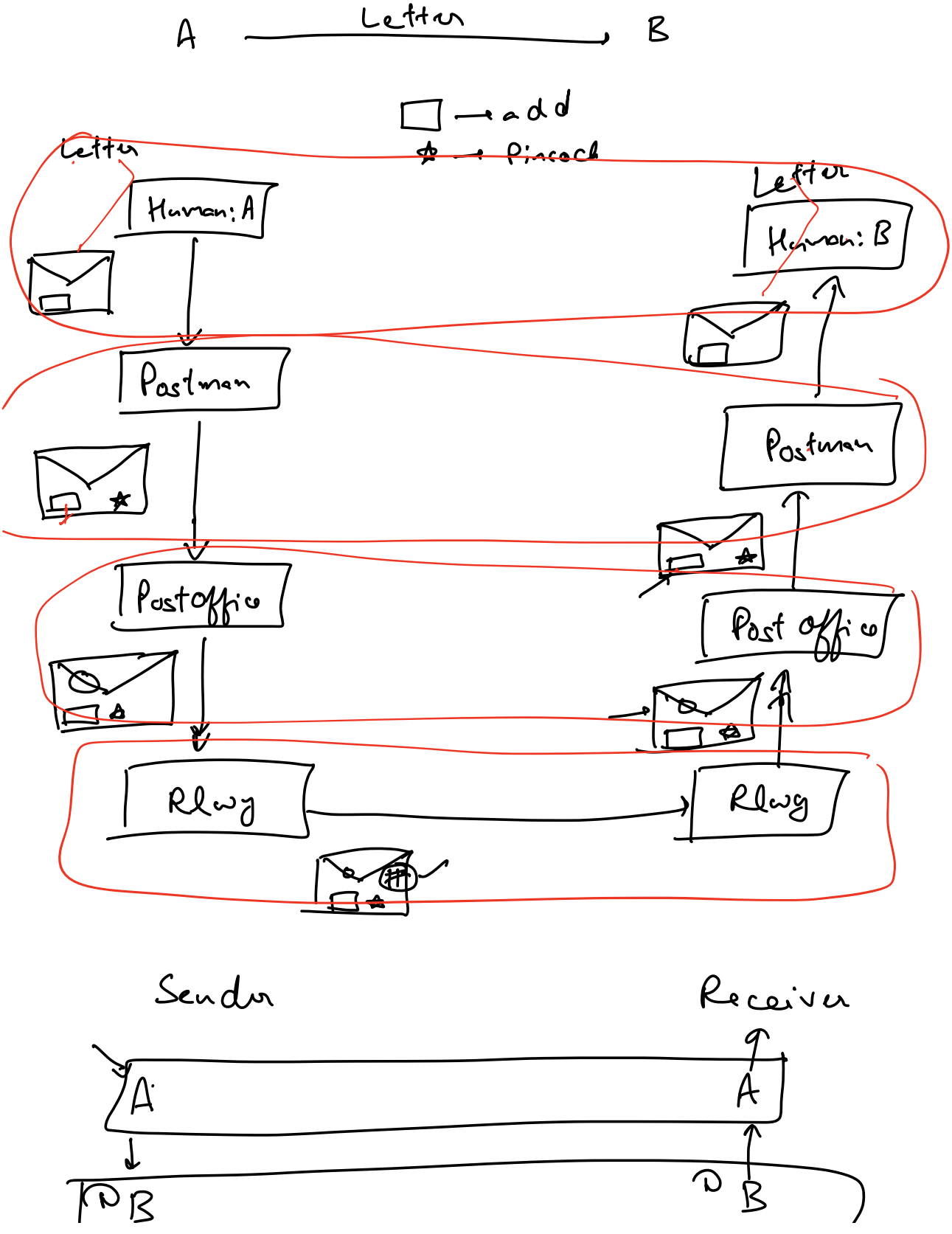
MySQL

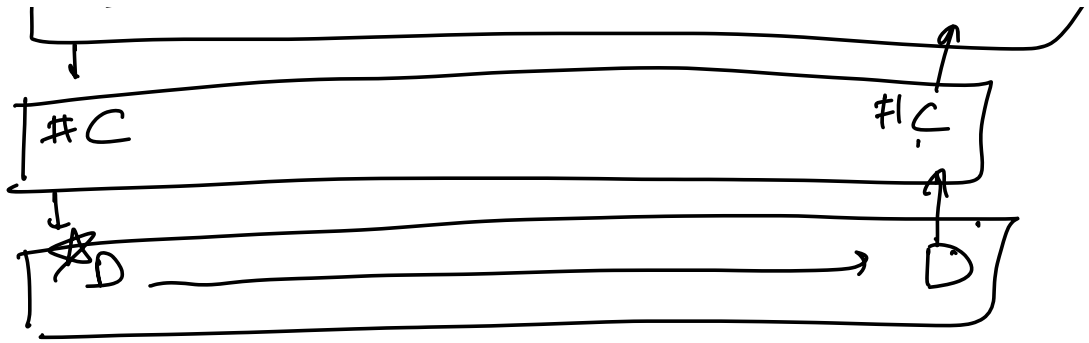
Layering Part (20 - 30 minutes)

[← HTTP / FTP]

[← TCP / UDP]

Architecture.





* Every layer takes info from previous layer. Add its info & passes on to next layer.

* Info added by a layer on sender's side is used by same layer on receivers side.

OSI Model : Theoretical

TCP/IP model

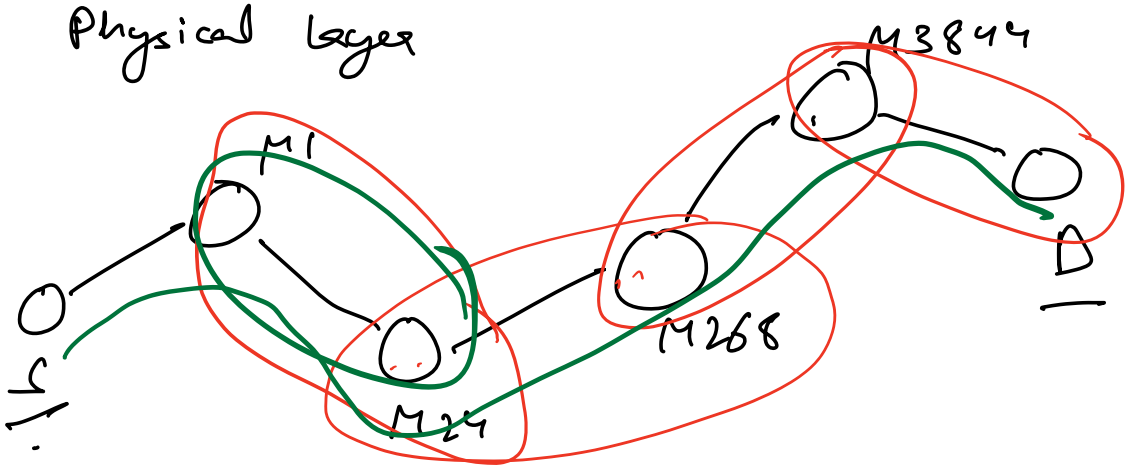
1. Application Layer (SW) → Take Input from user
2. Presentation layer (SW) → Encryption Compression
3. Session Layer (SW) → Adds user

4. Transport Layer → [Segmented, Rearranged, Checksum → corruption of data]

3. Network Layer → [Routing]

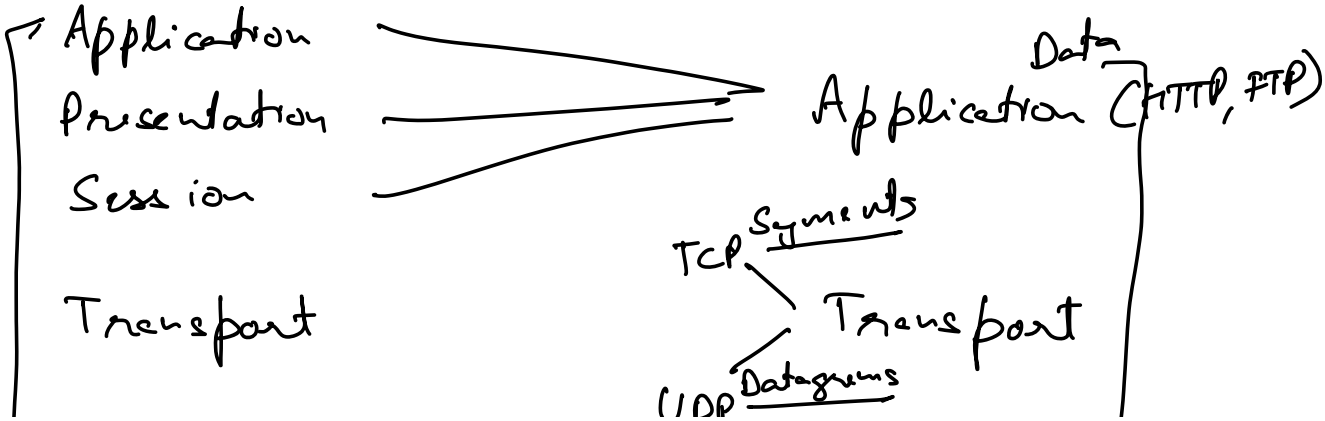
2. Data Link Layer → Hop to Hop communication

1. Physical Layer



OSI Model

TCP/IP Model



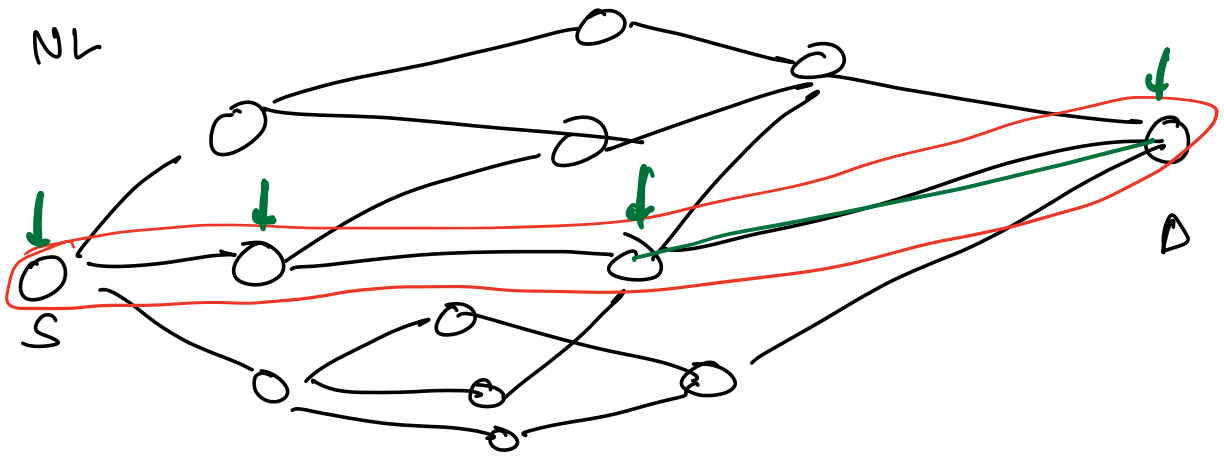
Network

Data Link

Physical Layer

IP — Packets — Network

Data Link — Frames



Application → HTTP, FTP

Transport → TCP → Guarantee

UDP → No Guarantee, Faster

VOLTE

Summary

IETF

- Where are protocals registered?
- IP addresses → ISP
- domain vs IP address → ICANN → DNS
- MAC Addresses → IEEE
  maintained by ISP
- Hardware Vendors → MAC addresses
- Ports identifies app on the machine
- Application layer protocals → http, ftp
  Transport layer " → TCP/UDP
  Network " " → IP
  Data Link "

OSI Model

App → Input

Presentation → Encryption, compression

TCP/IP Model

Application Layer

Session — User Info

Transport Layer → Segmentation
Checksum

Network Layer → Routing

Data Link Layer → Hop to Hop

Network Layer