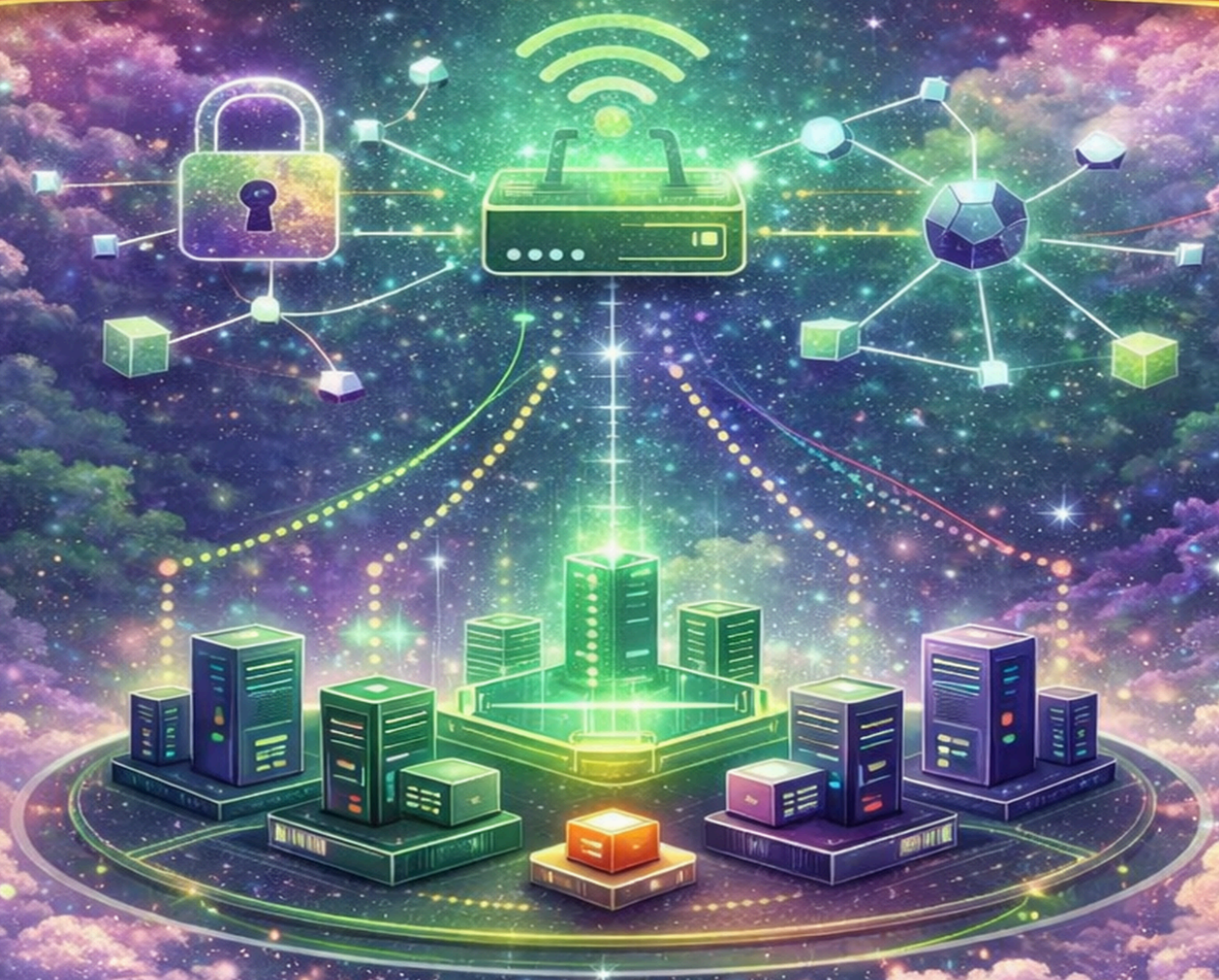
Networking Fundamentals for DEVOPS ENGINEERS



Learn With **Shubham Praharaj**

✅ Networking for DevOps (Complete Beginner → Advanced Notes)

Clean, structured, interview + real-world DevOps ready —

---

1) What is a Network?

A **network** is formed when **two or more computers/devices** are connected through **wired or wireless communication** so they can **share data and resources**.

Why Networks are Used

Networks help to:

- Enable **communication** between devices (messages, requests, data)
- Share **resources** like printers, storage, servers, and internet connection
- Support **centralized management** of systems
- Allow **remote access** and collaboration

✅ The **Internet** is the world's largest network (often called the **network of networks**).

---

2) Types of Networks

Networks are categorized based on size and coverage.

1. LAN (Local Area Network)

A network in a small area such as:

- Home
- Office
- School

✅ Examples: Ethernet, Wi-Fi

---

2. MAN (Metropolitan Area Network)

A network that spans a city or large campus.

✅ Example: City-wide ISP networks

---

3. WAN (Wide Area Network)

A large network connecting multiple LANs across long distances.

✅ Example: Internet, enterprise global networks
✅ Technologies: Fiber optic cables, leased lines

---

4. SONET (Synchronous Optical Network)

Used for high-speed **optical transmission**, especially long-distance and submarine communication networks.

✅ Example: Undersea internet cables

---

3) Common Networking Components

A DevOps engineer must understand basic networking hardware because cloud networking works on the same principles.

1. Switch

A **Layer 2** device that connects devices inside the same network and forwards traffic using **MAC addresses**.

✅ Used inside LAN networks
✅ Smarter than a hub (sends data only to the intended device)

---

2. Router

A **Layer 3** device that connects **different networks** and forwards traffic using **IP addresses**.

✅ Example: Connecting a private network to the internet

---

3. Modem

A device used for **modulation/demodulation**, converting signals for internet communication (mostly used in ISP connectivity).

---

4. Hub (Obsolete)

Broadcasts data to all ports (very inefficient and noisy).

✅ Replaced by switches

---

5. NIC (Network Interface Card)

A hardware component used to connect a device to a network.

✅ Every NIC has a unique **MAC Address**
✅ Now built into most motherboards (wired + wireless)

---

6. Bridge (Mostly Obsolete)

Connects two LANs and reduces unnecessary traffic compared to hubs.

✅ Replaced by switches

---

4) What is a Protocol?

A **protocol** is a set of rules that defines **how data is transmitted and received** between devices in a network.

✅ Examples:

- HTTP / HTTPS
- TCP / UDP
- IP
- DNS
- FTP
- SMTP / IMAP / POP3
- SSH

---

5) IP Address (Internet Protocol Address)

An **IP address** is a unique identifier given to a device on a network.

✅ Think of it as the **address of a device** so data can reach it correctly.

---

6) Types of IP Addresses

1. IPv4

- 32-bit address
- Format: `x.x.x.x`
- Example: `192.168.1.10`

✅ Total IPv4 addresses ≈ **4.3 billion**
⚠️ Running out due to internet growth (hence IPv6)

---

2. IPv6

- 128-bit address
- Written in hexadecimal groups
- Example:
  `2001:0db8:85a3:0000:0000:8a2e:0370:7334`

✅ Huge address space (solves IPv4 exhaustion)

---

3. Public IP

- Used on the internet
- Assigned by ISP/cloud provider
- Globally reachable

✅ Example: Your website server IP

---

4. Private IP

Used inside internal networks and **not routable on the public internet**.

✅ Private IPv4 ranges:

- `10.0.0.0/8`
- `172.16.0.0/12`
- `192.168.0.0/16`

✅ Example: EC2 private IP inside a VPC

---

5. Static IP

- Assigned manually or reserved
- Does not change

✅ Used for servers, database endpoints, VPN gateways

---

6. Dynamic IP

- Automatically assigned using DHCP
- Changes over time

✅ Used for laptops and home devices

---

7) IPv4 Classes (Important for Basics)

IPv4 was traditionally divided into classes:

Class	Range	Purpose
A	1.0.0.0 – 126.0.0.0	Large networks
B	128.0.0.0 – 191.255.0.0	Medium networks
C	192.0.0.0 – 223.255.255.0	Small networks
D	224.0.0.0 – 239.255.255.255	Multicast
E	240.0.0.0 – 255.255.255.255	Experimental

✅ Notes:

- `0.0.0.0` has special meaning (default route / unknown address)
- `127.0.0.0/8` is reserved for loopback

---

8) Loopback Address (Localhost)

A **loopback address** allows a device to communicate with itself.

✅ Common loopback:

- `127.0.0.1` → localhost

Why it's used

- Testing local services without external networking
- Debugging applications
- Checking if a service is running locally

---

9) Network ID vs Host ID

Every IPv4 address contains:

- **Network ID** → identifies the network
- **Host ID** → identifies the device inside that network

✅ Based on class:

- Class A: Network ID = 1st octet
- Class B: Network ID = 1st + 2nd octet

- Class C: Network ID = 1st + 2nd + 3rd octet

Connection rule

- Same Network ID → devices can talk directly
- Different Network ID → router is required

---

10) Subnetting (Core DevOps Skill)

Subnetting divides a network into **smaller networks** for:

- Security
- Performance
- Efficient IP usage

Example: Subnetting a /24 network

Network: `192.168.1.0/24`
Subnet Mask: `255.255.255.0`

✅ Total IPs: 256
✅ Usable IPs: 254 (network + broadcast excluded)

Split into two /25 networks

Subnet mask becomes: `255.255.255.128`

Subnet 1: `192.168.1.0/25`
Range: `192.168.1.0 – 192.168.1.127`
Usable: 126 hosts

Subnet 2: `192.168.1.128/25`
Range: `192.168.1.128 – 192.168.1.255`
Usable: 126 hosts

Benefits of Subnetting

✅ Reduces broadcast traffic
✅ Improves performance
✅ Improves isolation/security
✅ Helps structure cloud networking (public/private subnets)

---

11) CIDR (Classless Inter-Domain Routing)

CIDR replaced class-based networking and uses **prefix notation** like `/24`, `/16`.

✅ Format:
`IP/prefix`

Example:

- `10.0.0.0/16` → 65,536 addresses
- `192.168.1.0/24` → 256 addresses

Common CIDR blocks

Prefix	Netmask	Total Addresses	Meaning
/32	255.255.255.255	1	Single host
/24	255.255.255.0	256	Typical small subnet
/16	255.255.0.0	65,536	Large subnet

/8	255.0.0.0	16,777,216	Very large
/0	0.0.0.0	Entire internet	Default route

✅ `0.0.0.0/0` = **all destinations** (default internet route)

---

12) Network Models

Two major networking models are important in DevOps:

A) OSI Model (7 Layers)

The OSI model explains how data moves from one device to another.

OSI Layers (Top → Bottom)

1. **Application (L7)** – HTTP, FTP, SMTP
2. **Presentation (L6)** – SSL/TLS, Encryption, Compression
3. **Session (L5)** – Session management, NetBIOS
4. **Transport (L4)** – TCP, UDP
5. **Network (L3)** – IP, Routing, ICMP
6. **Data Link (L2)** – MAC Addressing, Ethernet, ARP
7. **Physical (L1)** – Cables, signals, radio waves

✅ Simple memory tip:
All People Seem To Need Data Processing

B) TCP/IP Model (4 Layers)

This is what is actually used in real networking.

TCP/IP Layers

1. **Application Layer** (OSI 7+6+5)
2. **Transport Layer** (OSI 4)
3. **Internet/Network Layer** (OSI 3)
4. **Network Interface Layer** (OSI 2+1)

✅ Real-world internet runs on TCP/IP.

---

13) Ports and Protocols (DevOps Must Know)

A **port** identifies a service running on a machine.

✅ Example:
IP = server address
Port = specific application/service on that server

Common Ports Cheat Sheet (Very Important)

Service	Protocol	Port
HTTP	TCP	80
HTTPS	TCP	443
SSH	TCP	22
DNS	UDP/TCP	53
SMTP	TCP	25
FTP	TCP	20/21
SFTP	TCP	22

MySQL	TCP	3306
PostgreSQL	TCP	5432
Redis	TCP	6379
Kubernetes API	TCP	6443

---

14) HTTP Basics (DevOps Perspective)

HTTP is a **client-server protocol** used for web communication.

✅ Important properties:

- Stateless by default (server does not store client session)
- Works with requests and responses

HTTP Methods (Most Common)

- `GET` → fetch data
- `POST` → send data
- `PUT` → update resource
- `DELETE` → remove resource

HTTP Status Codes (Most Common)

Success

- **200** OK

- **201** Created

Redirect

- **301** Permanent redirect
- **302** Temporary redirect

Client Errors

- **400** Bad request
- **401** Unauthorized
- **403** Forbidden
- **404** Not found

Server Errors

- **500** Internal server error
- **502** Bad gateway
- **503** Service unavailable
- **504** Gateway timeout

Cookies (Why login stays active)

Even though HTTP is stateless, login persists due to:
✅ **Cookies + Sessions + Tokens**

Cookies store identifiers in the browser so the server can recognize you.

---

15) TCP vs UDP

TCP (Transmission Control Protocol)

✅ Reliable + connection-based
✅ Used when delivery must be guaranteed

Examples:

- Web browsing (HTTPS)
- Database connections
- SSH

TCP Three-Way Handshake

1. SYN
2. SYN-ACK
3. ACK
   ✅ Connection established after this

UDP (User Datagram Protocol)

✅ Faster + connectionless
✅ No guaranteed delivery

Examples:

- DNS queries
- Video streaming
- Online gaming

---

16) Routing (How traffic moves between networks)

Routing decides how packets go from **source** to **destination** using a **route table**.

Routing Rule Priority

✅ Most specific route wins first
Example:

- `10.21.0.0/16` is evaluated before `10.0.0.0/8`
- Default route `0.0.0.0/0` matches everything else

Default Route Meaning

✅ `0.0.0.0/0` → send traffic to internet gateway / NAT / firewall

---

17) DNS (Domain Name System)

DNS converts:
✅ Domain name → IP address

Example:
`www.example.com` → `93.184.216.34`

How DNS Works (Simple Flow)

1. User types domain in browser
2. Query goes to resolver DNS
3. Resolver checks cache

4. If not found, asks:
    - Root DNS
    - TLD DNS (.com, .org)
    - Authoritative DNS
5. Returns IP to browser
6. Browser connects to server

✅ DNS is like the “phonebook” of the internet.

DNS Record Types (Most Common)

Record	Meaning	Use
A	IPv4 address	domain → IPv4
AAAA	IPv6 address	domain → IPv6
CNAME	alias	maps one name to another
MX	mail server	email routing
NS	name server	DNS delegation
TXT	text record	verification, SPF/DKIM

---

18) DHCP (Dynamic Host Configuration Protocol)

DHCP automatically assigns:

- IP address
- Subnet mask
- Gateway
- DNS servers

✅ Example: connecting your laptop to Wi-Fi and getting an IP automatically

---

19) Network Security & Services (DevOps View)

Firewalls

Firewalls control **allowed and blocked traffic** using rules.

✅ Examples:

- Allow SSH only from office IPs
- Block public access to internal databases

Load Balancer

A load balancer distributes traffic across multiple servers.

✅ Benefits:

- High availability
- Prevents overload
- Supports autoscaling

- Enables blue-green deployments

VPN (Virtual Private Network)

VPN creates a **secure encrypted tunnel** between users and a private network.

✅ Used for:

- Secure remote access
- Connecting offices
- Private cloud access

---

20) Network Troubleshooting Tools (Must Know for DevOps)

1. `ping`

Checks connectivity using ICMP.

```
ping google.com
```

✅ Useful for: packet loss + latency check

---

2. `traceroute` / `tracert`

Shows route path to destination.

```
traceroute google.com
```

✅ Useful for: identifying slow hops or failures

---

3. `telnet`

Checks if a port is reachable (basic connectivity test).

```
telnet example.com 443
```

✅ Useful for: port checks
⚠️ Not secure for remote login (use SSH instead)

---

4. `curl`

Used to test HTTP endpoints.

```
curl http://example.com
curl -I https://example.com
```

✅ Useful for: API testing + response headers

---

5. `dig`

DNS troubleshooting tool.

```
dig google.com
dig google.com A
```

✅ Shows TTL, records, authoritative servers

---

6. `netstat`

Shows active connections, listening ports, routing.

```
netstat -tulnp
netstat -r
```

✅ Useful for: port listening + routing check

---

7. `nmap`

Scans systems and ports.

```
nmap -sn 192.168.1.0/24
nmap -A 192.168.1.10
```

✅ Useful for: host discovery + port scanning
⚠️ Use only with permission

---

8. `ssh`

Secure remote login.

```
ssh user@server-ip
```

✅ Used daily in DevOps

---

9. `scp`

Securely copy files between machines.

```
scp file.txt user@server:/tmp/
```

✅ Great for quick transfers

✅ DevOps Networking Quick Summary (One Page Revision)

✅ Must-Know Concepts

- Public vs Private IP
- Subnetting + CIDR
- Routing tables + default route (0.0.0.0/0)
- DNS resolution
- TCP vs UDP
- OSI + TCP/IP models
- Security groups / firewall rules

✅ Must-Know Tools

- ping
- traceroute
- dig
- curl
- netstat
- ssh / scp
- nmap (permission-based use)