Exploring Network Protocols and Services

Network protocols

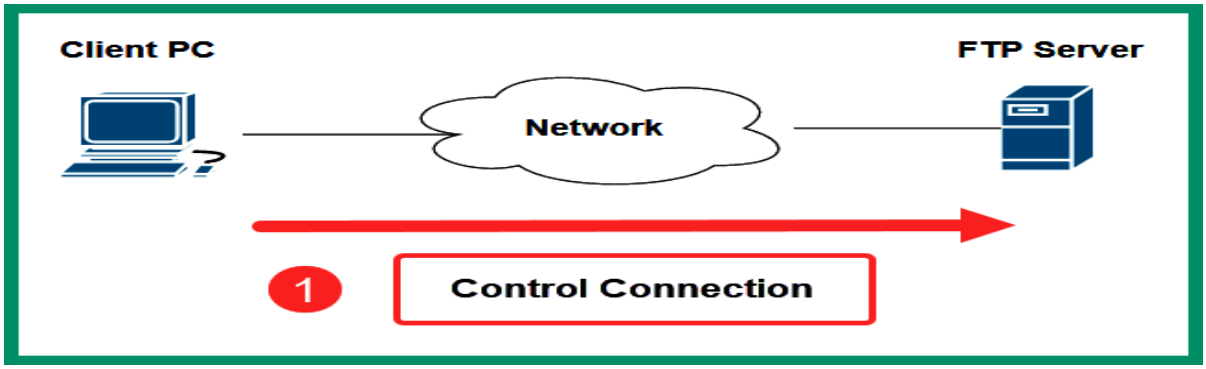
Network protocols are the **rules, standards, and procedures** that define **how devices communicate** over a network.
They specify **how data is packaged, addressed, transmitted, routed, and received**.

File protocols

1. File Transfer Protocol

FTP is a **file-sharing protocol** that allows a **client** (user) to connect to an **FTP server** to upload or download files over a network.

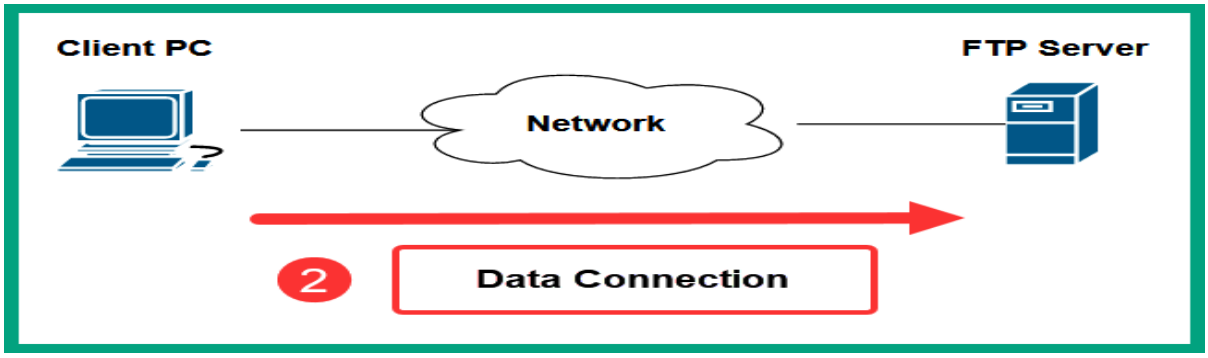
FTP works in a **client–server model** and uses **two separate TCP connections**:



1. Control Connection (Port 21)

- This connection is used for **sending commands** (e.g., login, list files, change directory).
- It remains **open during the entire FTP session**.
- Uses **TCP port 21** on the server.

Purpose: To control and manage the session.

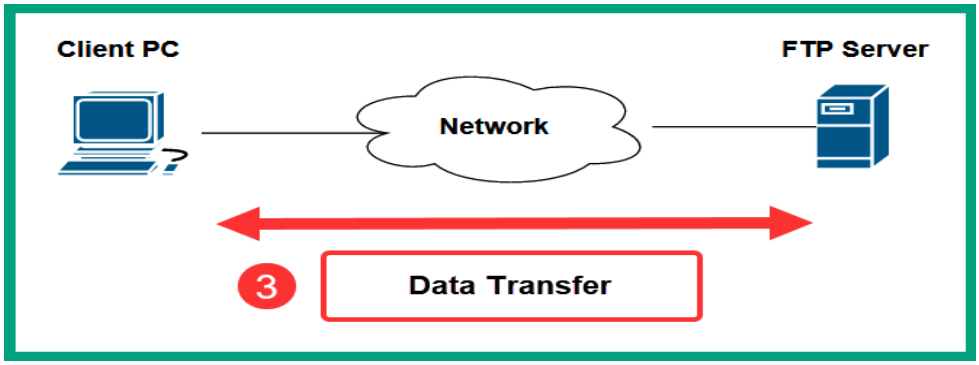


2. Data Connection (Port 20)

(Shown as step **2**)

- Used to **actually transfer files or directory listings**.
- Uses **TCP port 20** on the server.
- It opens only when needed (e.g., during upload/download).

Purpose: To carry the actual data.



3. Data Transfer

(Shown as step **3**)

Once both control and data connections are set up:

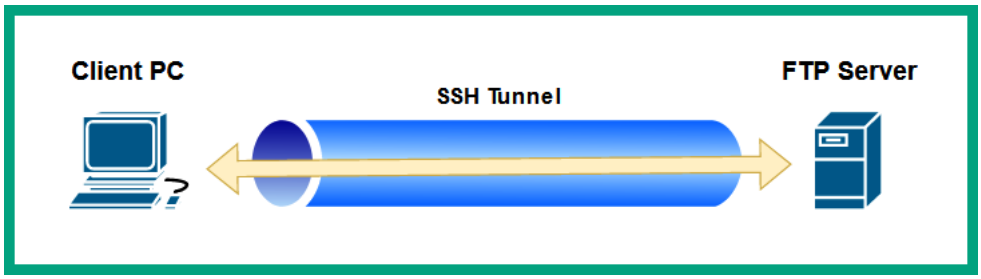
- Files can be **uploaded** to the server
- Files can be **downloaded** from the server
- Data moves **both ways** over the data connection.

2. `SSH File Transfer Protocol

SFTP is a **secure method** of transferring files between a client and a server.

Unlike traditional FTP, which sends data in **plaintext**, SFTP sends all data through an **encrypted SSH tunnel**. This protects the data from hackers.

How SFTP Works



1. **Client opens an SSH connection** to the server
    - Uses **TCP port 22** (SSH port)
    - This connection is fully **encrypted**
2. **An SSH tunnel is created**
    - All communication between client and server now passes through this secure tunnel
    - No one on the network can read the data
3. **File transfer happens inside the tunnel**
    - FTP-like commands and file data are sent **inside the encrypted SSH session**
    - Even if someone intercepts the packets, they will only see **encrypted data**

3. File Transfer Protocol Secure (FTPS)

FTPS (FTP Secure), also called **FTP over SSL/TLS**, is an extension of FTP that uses **SSL or TLS** to encrypt file-transfer communication between a client and server.

How FTPS Works

- The client encrypts **each FTP message** using SSL/TLS.
- Unlike SFTP (which creates a secure SSH tunnel), FTPS **does not use a tunnel**.
- Every FTPS packet is **individually encrypted** before being sent over the network.
- The FTPS server **decrypts each message** and reassembles the data.

FTPS Ports

FTPS can operate in **two modes**:

✓ 1. Implicit FTPS – Port 990

- Client connects to the server on **port 990**.
- SSL/TLS encryption is **expected immediately**.
- Secure connection starts automatically.

✓ 2. Explicit FTPS – Port 21

- Client connects to the server on **port 21** (same as FTP).
- Client must explicitly request security by sending:
    - **AUTH SSL**, or

- **AUTH TLS**
- After this, SSL/TLS handshakes begin.
- Security is optional → good for mixed environments.

4. Trivial File Transfer Protocol (TFTP)

Definition:
TFTP (Trivial File Transfer Protocol) is a **simple, lightweight, connectionless** version of FTP.
Used mainly for transferring files to network devices (routers, switches).

TFTP uses UDP port 69

Difference Between FTP and TFTP

Feature	FTP (File Transfer Protocol)	TFTP (Trivial File Transfer Protocol)
Full Form	File Transfer Protocol	Trivial File Transfer Protocol
Protocol Type	Full-featured, reliable file transfer	Simple, lightweight file transfer
Transport Protocol	**TCP**	**UDP**
Port Number	TCP **20 (data)** and **21 (control)**	UDP **69**
Connection Type	Connection-oriented (3-way handshake)	Connectionless (no handshake)
Security	Supports authentication (username/password)	No authentication, not secure
Features	Upload, download, list directories, delete, rename files	Basic upload & download only
Use Case	User file sharing across networks	Network device config/OS transfer (routers & switches)
Complexity	More complex, more overhead	Very simple, low overhead
Speed	Slower than TFTP due to reliability checks	Faster but less reliable

5. Server Message Block (SMB)

Server Message Block (SMB) is a **client–server file-sharing protocol** primarily used in **Microsoft Windows environments**.
It allows devices on a network to access shared files, folders, printers, and other network resources.

SMB enables **network-based resource sharing** similar to a local file system experience.

Purpose of SMB

SMB allows:

- File sharing
- Printer sharing
- Communication between applications
- Network browsing of shared resources

It is widely used for:

- Windows file servers
- Domain environments
- Shared directories and network drives (e.g., `\\Server\Share`)

Functions of SMB

✓ 1. Session Management

SMB handles:

- Starting sessions
- Authenticating users
- Terminating sessions

This ensures secure and controlled access between the client and the server.

✓ 2. File and Printer Access Control

SMB manages:

- Opening, reading, writing, and closing files
- Sharing printers across networks
- Permission controls (read, write, execute)

SMB ensures only authorized users can access shared files or printers.

✓ 3. Application Messaging

SMB allows applications to:

- Exchange data
- Send control messages

SMB Ports

- Modern SMB (SMB 2 & SMB 3) uses **TCP port 445**
- Older versions (SMB over NetBIOS) use:
  - UDP 137 & 138
  - TCP 137 & 139

Port 445 is the primary port for current Windows systems.

Remote access protocols

1. Telnet

Telnet is an old (legacy) **remote access protocol** that allows IT professionals to **remotely connect** to devices such as:

- Computers
- Servers
- Routers and switches
- Security appliances
- IoT devices

• Telnet sends **all data in plaintext**:

  - Usernames
  - Passwords
  - Commands
  - Output

- Attackers can easily capture Telnet traffic using sniffing tools.
- No encryption, no confidentiality, no integrity. Because of this, Telnet is NOT recommended on modern networks.
- Telnet uses TCP port 23

2. Secure Shell (SSH)

Secure Shell (SSH) is a **secure remote access protocol** used by IT and network professionals to remotely connect to devices such as:

- Routers
- Switches
- Servers
- Firewalls
- IoT devices

SSH uses TCP port 22 by default

Why SSH Is Important

SSH is the secure replacement for Telnet.

Unlike Telnet, **SSH encrypts all communication**, including:

- Username
- Password
- Commands entered
- Output returned

This prevents attackers from reading sensitive information even if they intercept the traffic.

How SSH Protects Communication

SSH ensures:

✓ Confidentiality

All data is encrypted, making it unreadable to attackers.

✓ Integrity

Data cannot be altered in transit without detection.

✓ Authentication

Supports:

- Password-based authentication
- Key-based authentication (more secure)

Even if a hacker intercepts SSH packets, the encrypted data **cannot be decrypted** without the key.

Difference between Telnet and SSH

Feature	Telnet	SSH
Meaning	A remote access protocol	A secure remote access protocol
Security	✗ No encryption	✓ Encrypted end-to-end
Port	TCP **23**	TCP **22**
Data Protection	✗ Plain text	✓ Secure, encrypted
Best Used For	Old, trusted networks	Modern secure networks
File Transfer	✗ No	✓ Yes (SCP, SFTP)

3. Remote Desktop Protocol (RDP)

Remote Desktop Protocol (RDP) is a **secure remote access protocol** developed by Microsoft.
It allows IT professionals to **remotely access and control Windows computers and servers**.RDP is **built into all modern Windows operating systems**, so no extra software is required on Windows machines.

Service port no 3389

Email protocols

1. Simple Mail Transfer Protocol (SMTP)

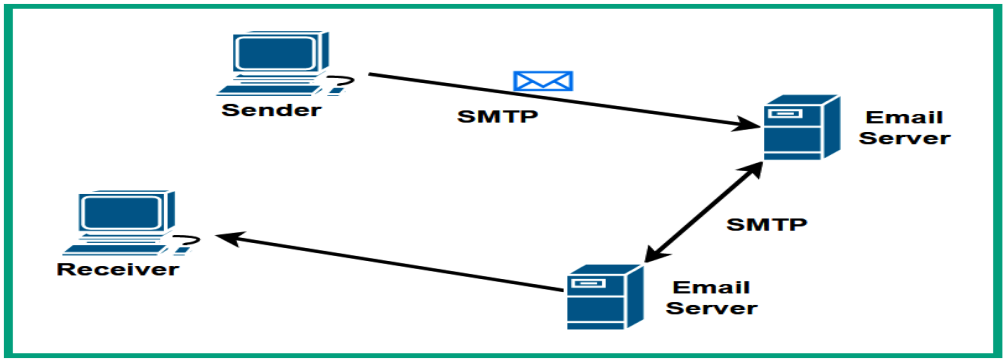
- Used to **send** email messages, not retrieve them.

- Uses **TCP port 25 by default**.

- SMTP handles:

1. Client → Sender's mail server
2. Sender's mail server → Recipient's mail server
3. Final delivery toward recipient

Final delivery toward recipient



2. Post Office Protocol (POP)

The **Post Office Protocol (POP)** is an email protocol used by email clients (such as Microsoft Outlook, Thunderbird, Apple Mail) to **download emails** from an email server to a local device.

POP is mainly designed for **offline email access**.

POP mail servers **listen on TCP port 110** for incoming connections from clients.

This is the default port used by POP3 (current version).

How POP Works

Step 1: Email client connects to server

- The email client establishes a connection to the mail server using:
  - **TCP port 110**
- The server waits ("passively listens") for POP connections.

Step 2: Client downloads emails

- After authentication, the POP client **downloads all emails** from the user's mailbox on the server.

Step 3: Emails removed from server

- Once downloaded, emails are **deleted from the server** by default.
- This means the master copy of your emails is stored **locally** on your computer

3. Internet Message Access Protocol (IMAP)

IMAP (Internet Message Access Protocol) is an email protocol used by email clients (like Outlook, Thunderbird, Gmail app) to **synchronize** email messages between the **email server** and the **client device**.

- It allows you to view and manage your emails **without downloading and deleting them from the server**.
- **IMAP uses TCP port 143**
- Secure IMAP (IMAPS) uses **TCP port 993** (SSL/TLS encrypted)

Short questions on Email protocols

What is SMTP and why is it used?

SMTP (Simple Mail Transfer Protocol) is used for **sending emails** from a client to a mail server and between mail servers. It operates on ports **25/465/587**.

Differentiate between POP3 and IMAP.

POP3 **downloads** emails to a client and usually **deletes** them from the server, whereas IMAP **synchronizes** emails and keeps messages on the server for multi-device access.

What is the function of the POP3 protocol?

POP3 (Post Office Protocol v3) is used to **retrieve and download emails** from a mail server to a single device.

What role does IMAP play in email communication?

IMAP (Internet Message Access Protocol) allows users to **access, synchronize, and manage emails stored on a server** from multiple devices.

State two differences between SMTP and POP3.

1. SMTP is used to **send** emails; POP3 is used to **receive** emails.
2. SMTP works between mail servers, while POP3 works between a **client and server**.

Mention any two default port numbers used in email protocols.

1. SMTP: **25/465/587**
2. POP3: **110/995**
3. IMAP: **143/993**

Why is IMAP preferred over POP3?

IMAP allows **server-based email storage**, **folder management**, and **multi-device synchronization**, making it more flexible than POP3.

Explain the purpose of email client software.

Email clients like Outlook or Thunderbird enable users to **send, receive, store, and manage emails** using protocols such as SMTP, POP3, and IMAP.

What is the role of an email server?

An email server stores incoming messages, manages user mailboxes, and routes outgoing emails using protocols like **SMTP, POP, IMAP**.

What is the difference between port 110 and port 995?

Port **110** is used for **POP3 (unencrypted)** communication, while port **995** is used for **POP3S (POP3 secured with SSL/TLS)**.

Give any two advantages of IMAP.

1. Emails stay on the server.
2. Can access same mailbox from multiple devices.

HTTP

HTTP (Hypertext Transfer Protocol) is an application-layer protocol used by web browsers and web servers to communicate.

✓ Key Features of HTTP

- **Unsecure protocol**
  - Does **not** provide confidentiality, integrity, or authentication.
  - Data is transmitted in **plaintext**, which can be intercepted by attackers.
- **Default port: 80**
- **Used for:**
  - Webpage browsing
  - Downloading resources such as images, text, and HTML files

HTTPS

HTTPS (Hypertext Transfer Protocol Secure) is the secure version of HTTP.

Key Features of HTTPS

- **Provides encryption**, which protects confidentiality.
- Uses **SSL/TLS** (Secure Sockets Layer / Transport Layer Security).
- Authenticates the web server using a **digital certificate** issued by a Certificate Authority (CA).
- **Default port: 443**
- Ensures:
  - ✓ Confidentiality
  - ✓ Integrity
  - ✓ Authentication

Difference between HTTP and HTTPS

Feature	HTTP	HTTPS
Full Form	Hypertext Transfer Protocol	Hypertext Transfer Protocol Secure
Security	✗ No encryption	✓ Encrypted using SSL/TLS
Default Port	80	443
Server Authentication	✗ Not provided	✓ Provided using digital certificates
Data Protection	✗ Plaintext	✓ Confidential & encrypted
Use Cases	General web browsing	Secure transactions, login pages

Network protocol types

1. Internet Control Message protocol

ICMP is a network-layer protocol used to send error messages and diagnostic information. It supports tools like PING and traceroute for testing connectivity.

Type	Name	Code
0	Echo Reply	0
3	Destination Unreachable	0 - Nework Unreachable
		1 - Host Unreachable
		2 - Protocol Unreachable
		3 - Port Unreachable
		4 - Fragmentation needed and "Don't Fragment" was set
5	Redirect	0 - Redirect for the Network
		1 - Redirect for the Host
8	Echo Request	0
11	Time Exceeded	0 - Time to Live (TTL) exceeded
		1 - Fragment reassembly time exceeded

Figure ICMP codes and types

ICMP messages are identified by:

- **Type** = What kind of message it is
- **Code** = More specific reason within that type

Type 0 – Echo Reply

- **Used in:** PING responses
- **Meaning:** The destination host is alive and responding.
- **Code 0:** Standard reply

Example:
You ping 8.8.8.8 → it sends an **Echo Reply** (Type 0) back.

✓ Type 3 – Destination Unreachable

This message means **the packet cannot reach its destination**.
The **Code** tells the exact reason.

Codes under Type 3:

Code	Meaning	Explanation
0 – Network Unreachable	Router cannot find the network	No route to the network prefix
1 – Host Unreachable	Destination host down / unreachable	Device offline or ARP failure
2 – Protocol Unreachable	Protocol not supported	Example: sending a UDP packet to a host that doesn't support it
3 – Port Unreachable	Target port closed	Common for UDP; host responds that port is closed
4 – Fragmentation Needed (DF Set)	Packet too large & cannot fragment	Happens when MTU is small and DF bit is set

✓ Example:

If you send UDP to a closed port, the destination sends:
Type 3, Code 3: Port Unreachable

✓ Type 5 – Redirect

Routers send this message to inform a host that it should use a **different gateway**.

Codes under Type 5:

Code	Meaning
0 – Redirect for Network	Use another router for the network
1 – Redirect for Host	Use another router for this host

✓ Example:

If a host sends packets to Router A, but Router B is the better path, Router A sends an **ICMP Redirect**.

✓ Type 8 – Echo Request

- Sent by the **PING** command.
- Used to check if a host is reachable.
- Always paired with Type 0 (Echo Reply).

✓ Example:

When you ping a device, your system sends:
Type 8, Code 0 → Echo Request

✓ Type 11 – Time Exceeded

This message is sent when the packet **expires before reaching destination**.

Codes under Type 11:

Code	Meaning
0 – TTL Exceeded	TTL became 0 before reaching destination
1 – Fragment Reassembly Time Exceeded	Host took too long to reassemble fragments

✓ Example:

Used in **traceroute**.
Each router where TTL becomes 0 sends:
Type 11, Code 0 → Time Exceeded

2. TCP (Transmission Control Protocol)

Defined by: RFC 793
Layer: Transport Layer (OSI Layer 4 / TCP-IP Transport Layer)
Type: Connection-oriented protocol

✓ Key Characteristics

- Establishes a **TCP 3-way handshake** (SYN → SYN-ACK → ACK) before data transfer.
- Provides **reliable data delivery**:
  - Every segment sent must be **acknowledged (ACK)** by the receiver.
  - If ACK is not received within a timeout, the sender **retransmits** the segment.
- Ensures:
  - **Error checking**
  - **Flow control** (using window size)
  - **Congestion control**
  - **In-order delivery** of data
- Used for applications that require accuracy.

✓ Use Cases

- Web browsing (HTTP/HTTPS)
- Email (SMTP, IMAP)

- File transfer (FTP)
- Remote login (SSH, Telnet)

3. UDP (User Datagram Protocol)

Defined by: RFC 768
Layer: Transport Layer
Type: Connectionless protocol

✓ Key Characteristics

- **No connection setup** (no handshake).
- **No guarantee of delivery**:
  - No acknowledgments
  - No retransmissions by UDP itself
- Fast and low-overhead:
  - Better for time-sensitive applications
- If packets are lost, the **application layer** must handle retransmission (not UDP).

✓ Use Cases

- Live streaming (video/audio)
- Online gaming
- VoIP (voice over IP)

Network Services

As networks grow, organizations rely on several **network services** to ensure smooth communication and proper functioning of devices.
Common network services include:

- Time synchronization
- Automatic IP address assignment
- Logging
- Directory services
- Name resolution (DNS)
- Remote access services

Network Time Protocol (NTP)

NTP (Network Time Protocol) is a protocol that allows devices on a network to **synchronize their system clocks** to the same, accurate time.

Port and Transport

- Runs on UDP port 123

- Operates in a client–server model

NTP is Important

✓ 1. Consistent time across all devices

All computers, servers, routers, firewalls, and switches have the **same system time**.

✓ 2. Automated tasks run correctly

Examples:

- Backups
- Updates
- Scripts
- Batch jobs

If device clocks are incorrect, these tasks may **run at the wrong time or fail altogether**.

✓ 3. Accurate timestamps in logs (Syslog)

Syslog messages include:

- Date
- Time
- Event information

If devices have mismatched clocks:

- Logs from different devices won't match
- Sequence of events becomes unclear
- Troubleshooting becomes difficult

✓ 4. Security and incident analysis

Accurate time helps identify:

- When an attack happened
- What systems were affected
- Exact order of events

Security Note

NTP is an **unsecure protocol** and can be exploited by attackers (e.g., amplification attacks). However, NTP supports **authentication**, allowing clients to verify the identity of the NTP server.

Dynamic Host Configuration Protocol (DHCP)

DHCP is a widely used **network protocol and service** that allows network administrators to **automatically assign IP addressing information** to client devices on a network.

When a device such as a **laptop, PC, smartphone, printer, or IoT device** connects to a network, it requires the following:

- **IP address**
- **Subnet mask**
- **Default gateway**
- **DNS server address**

Manually assigning this to every device is slow, inefficient, and causes errors like **duplicate IP addresses**.
DHCP solves this by **automating IP address distribution**.

✓ Why DHCP Is Needed

Without DHCP, each device must be manually configured with an IP address. In large organizations, this is:

- ✗ Time-consuming
- ✗ Error-prone
- ✗ Hard to manage
- ✗ Likely to cause IP conflicts

With DHCP:

- ✓ Devices automatically receive IP configurations
- ✓ No duplication
- ✓ Simple network management
- ✓ Faster setup for new devices

✓ What DHCP Provides

DHCP gives client devices:

1. **IP Address**
2. **Subnet Mask**
3. **Default Gateway**
4. **DNS Server**
5. **Lease Time**

These allow the device to communicate on the network and access external services.

✓ DHCP Server Configuration Options

Scope: The range of IP addresses (pool)
Exclusion ranges: The IP addresses that should not be distributed on the network
Reservation: Reserves IP addresses from the pool
Dynamic assignment: Dynamically assigns an IP address to a client on the network
Static assignment: Statically configures an IP address on a client
Lease time: Sets the time that the client can use the IP address given from the DHCP server
Scope options: Additional operations that can be configured when creating the scope
Available leases: Identifies the available lease time for an IP address

Explanation

Network professionals configure several components on a DHCP server:

✓ 1. Scope

A **scope** defines the **range of IP addresses** (address pool) that DHCP will assign to clients.
Example: 192.168.1.50–192.168.1.200

✓ 2. Exclusion Ranges

These are IP addresses **not to be assigned** by DHCP because they are used by:

- Servers
- Routers
- Printers
- Static devices

Example: Excluding 192.168.1.1–192.168.1.20

✓ 3. Reservation

A specific device (identified by MAC address) always receives the **same IP**.
Useful for printers, servers, CCTV cameras, etc.

✓ 4. Dynamic Assignment

DHCP assigns an IP address **automatically from the pool** whenever a client joins the network.

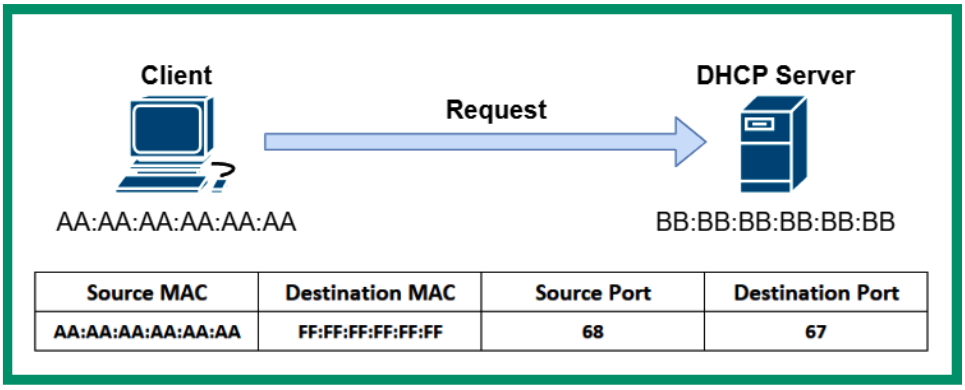
✓ **5. Static Assignment (Manual)**

Admin manually assigns a specific IP to a specific device.
(Not dynamically assigned by DHCP.)

DHCP 4-way handshake

DHCP client sends a DHCP message from a source service port of 68. The DHCP server operates on service port 67 by default.

1. The client connects to the network and sends a DHCP `Discover` message, seeking a DHCP server on the network

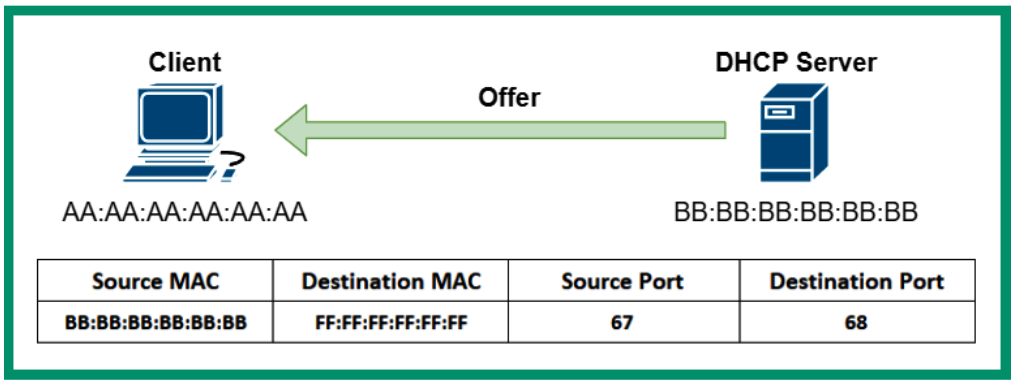


Source MAC	Destination MAC	Source Port	Destination Port
AA:AA:AA:AA:AA:AA	FF:FF:FF:FF:FF:FF	68	67

DHCP Discover message

As shown in the preceding diagram, the client includes its source **Media Access Control** (**MAC**) address and source port number, which is 68.
The client inserts the destination MAC address as FF:FF:FF:FF:FF:FF with the broadcast MAC address for a Layer 2 network and the destination port number, which is 67. The source IP address on the packet is left blank while the destination IP address is set to 255.255.255.255.

2. Next, the DHCP server responds with a DHCP **Offer** message, which contains the IP address needed by the client for communication on the network:

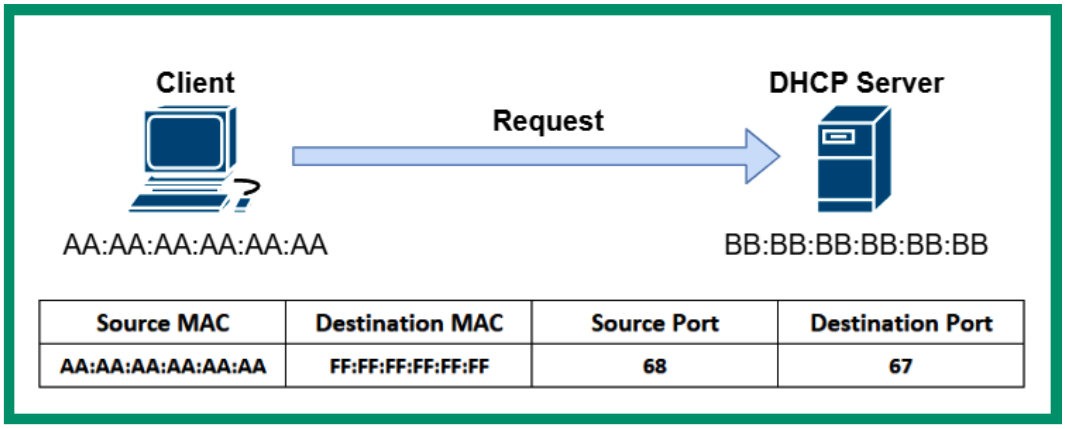


Source MAC	Destination MAC	Source Port	Destination Port
BB:BB:BB:BB:BB:BB	FF:FF:FF:FF:FF:FF	67	68

DHCP Offer message

As shown in the preceding diagram, the DHCP Offer message is sent from the DHCP server to the client on the network. The DHCP Offer message contains the source MAC address and source port number 67 of the DHCP server. This message also contains the destination MAC and destination port number 68 of the client on the network.
The DHCP Offer message is sometimes sent as unicast or broadcast to the client on the network.

3. Next, the client sends a DHCP Request message to the DHCP server, indicating that it will use the IP addresses from the previous message:

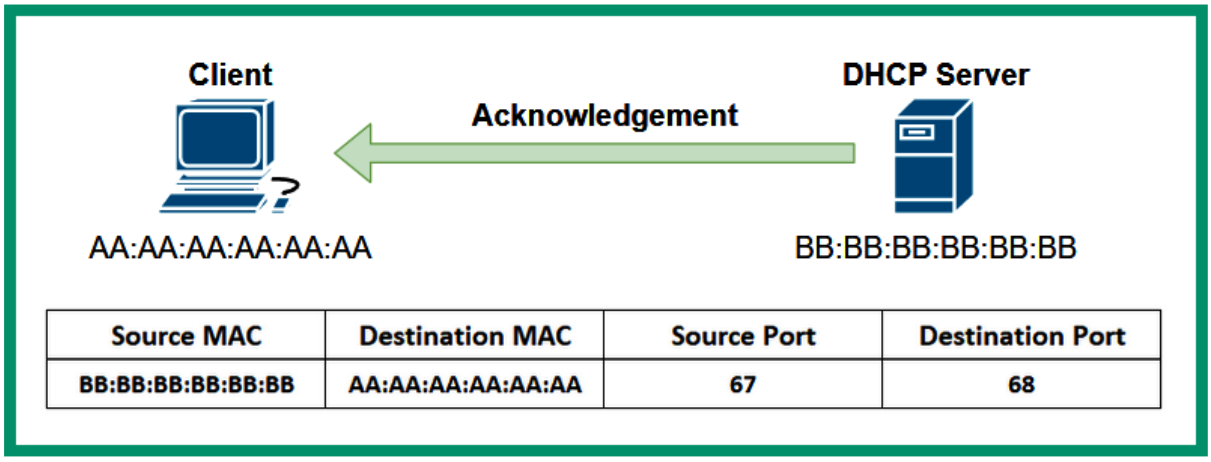


Source MAC	Destination MAC	Source Port	Destination Port
AA:AA:AA:AA:AA:AA	FF:FF:FF:FF:FF:FF	68	67

DHCP Offer message

As shown in the preceding diagram, the client sends a Layer 2 broadcast message to the DHCP server on the network, even though the client knows the MAC address of the DHCP server from the DHCP **Offer** packet.

4. Lastly, the DHCP server responds with a DHCP **Acknowledgment** unicast message to confirm the client can use the IP address provided from the addressing pool on the server:



Source MAC	Destination MAC	Source Port	Destination Port
BB:BB:BB:BB:BB:BB	AA:AA:AA:AA:AA:AA	67	68

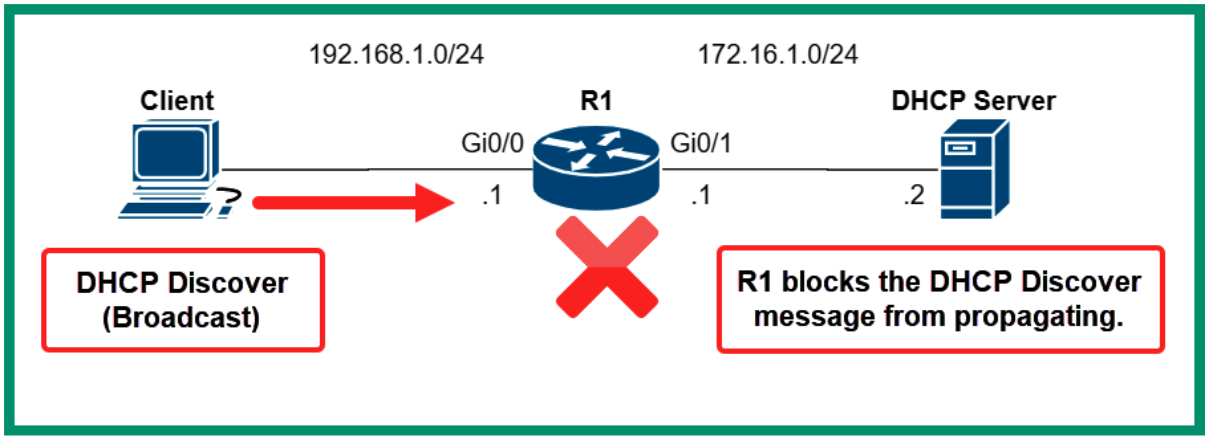
DHCP Acknowledgement message

The client is allowed to use the IP addresses provided by the DHCP server for the duration of the lease. If the client wants to extend the lease of communication on the network, the client can send a DHCP **Request** (unicast) message to the DHCP server to request the renewal of the lease.

If a client connects to a network but the DHCP server is on a **different IP subnet**, the client cannot reach it directly. This is because the client sends a **DHCP Discover** message, which is a **broadcast** message. Broadcasts only stay within the **same Layer 2 network** and **cannot pass through a router**.

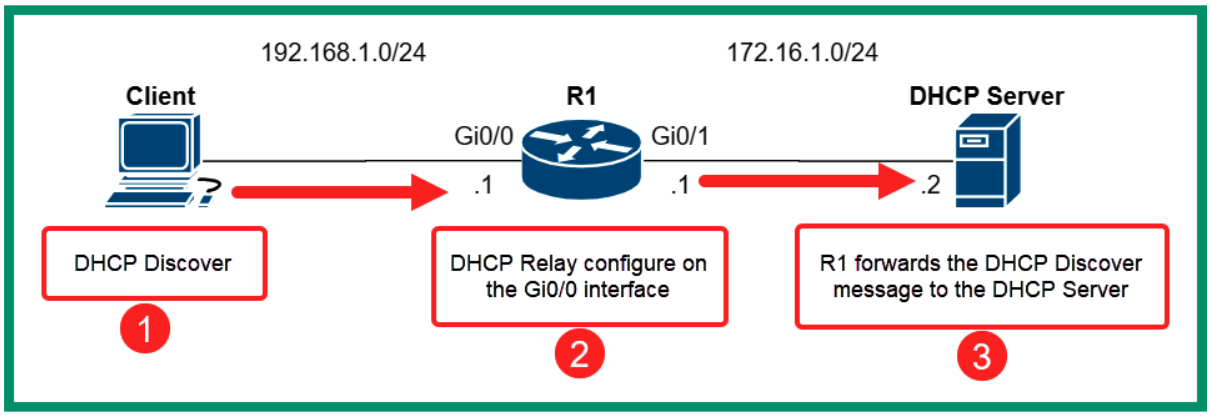
So, the router will block the DHCP Discover message, and the client will not get an IP address unless a **DHCP Relay (IP helper address)** is configured on the router.

The following diagram shows a router blocking a **DHCP Discover** message from propagating to another network:



Routers and other Layer 3 devices **do not forward Layer 2 broadcast messages**. This means a client's **DHCP Discover** broadcast cannot reach a DHCP server on a different network.

To fix this, the router must be configured as a **DHCP Relay Agent**. A DHCP Relay forwards DHCP messages between clients and servers across different networks so the client can receive an IP address even if the server is on another subnet.



DHCP Relay agent

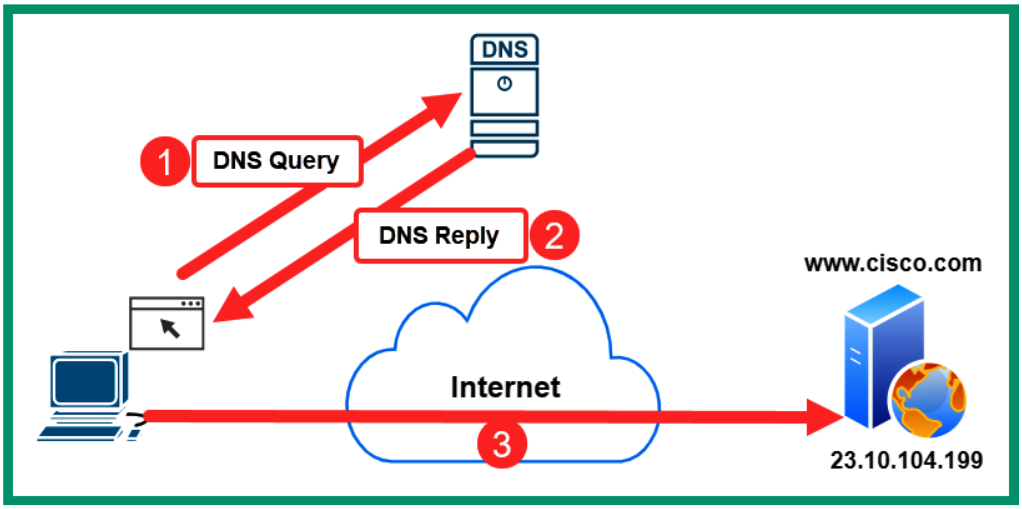
Domain Name System

DNS (Domain Name System) is a protocol that translates a **Fully Qualified Domain Name (FQDN)** like *www.cisco.com* into an **IP address**.
It works like a **telephone directory**, storing and mapping hostnames to IP addresses.

Why DNS is Needed

- Humans remember names better than numbers.
- Servers may change IP addresses; DNS keeps the hostname consistent.
- Clients use DNS to find the IP address of websites, servers, and services.

How DNS Works (Simple Flow)



1. The client wants to access **www.cisco.com** but does not know the IP address.
2. It sends a **DNS Query** to the DNS server on **UDP port 53**.
3. The DNS server looks up its database for the domain.
4. It sends back a **DNS Reply** with the correct IP address.
5. The client uses that IP address to connect to the website.

✓ DNS uses **UDP 53** for queries.
✓ DNS uses **TCP 53** for zone transfers between DNS servers.

Common DNS Record Types and Their Purpose

1. A Record

- Maps a hostname to an **IPv4** address.

2. AAAA Record

- Maps a hostname to an **IPv6** address.

3. CNAME (Canonical Name)

- Maps an **alias** to a real domain name.
  Example: *mail.example.com → server1.example.com*

4. MX (Mail Exchange)

- Specifies the **mail servers** for a domain.

5. SOA (Start of Authority)

- Defines the **primary DNS server** and domain authority information.

6. PTR (Pointer Record)

- Maps an **IP address back to a hostname** (reverse DNS lookup).

7. TXT Record

- Stores text information, often used for **domain verification** (Google, Microsoft, SPF).

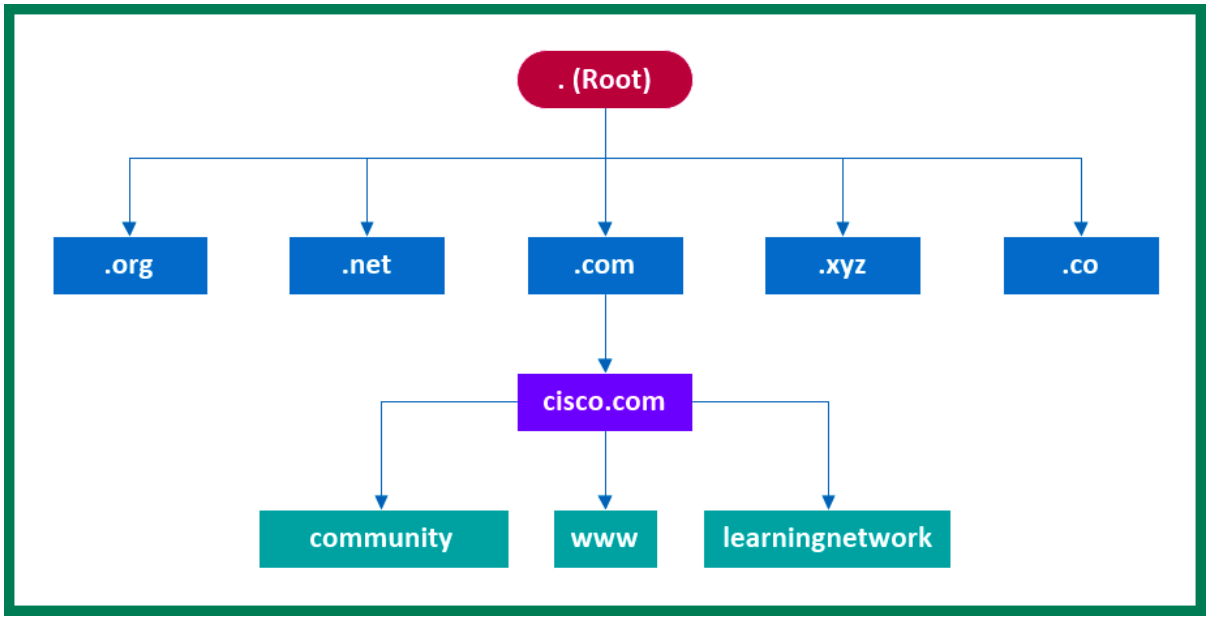
8. SRV Record

- Stores information about **specific services** for the domain (VoIP, AD services).

9. NS Record

- Lists the **name servers** responsible for the domain.

- An **authoritative DNS server** is the final holder of an IP address for a domain name or hostname on a network.
- The authoritative DNS server for a domain contains the original DNS records that are associated with a domain.
- However, the **recursive DNS server** or **non-authoritative DNS server** does not hold the original DNS records for a domain but queries an authoritative server when needed


. (Root)
.org
.net
.com
.xyz
.co
cisco.com
community
www
learningnetwork