**Bootstrap Protocol (BOOTP)** is a computer networking protocol used in **Internet Protocol (IP) networks** to automatically assign an **IP address** to network devices from a configuration server.

When a computer connected to a network is **powered on and boots its operating system**, the system software **broadcasts BOOTP request messages** on the network to obtain an IP address. A **BOOTP configuration server** listens to these requests and assigns an IP address to the device from a **predefined pool of addresses** configured by a network administrator.

**How a BOOTP Configuration Server Assigns an IP Address**

1. **Client boots without an IP**
    - When a device is powered on, it does **not have an IP address**.
    - It knows only its **MAC (hardware) address**.
2. **BOOTP request is broadcast**
    - The client sends a **BOOTP Request** as a **broadcast message** to the network.
    - This message contains:
        - Client **MAC address**
        - Request for IP configuration
3. **BOOTP server listens**
    - The BOOTP server is always running and **listens for BOOTP requests** on the network.
    - It receives the broadcast request from the client.
4. **Server checks its configuration table**
    - The server has a **predefined table** created by the network administrator.
    - This table maps:
        - **MAC address → IP address**
    - Example:
        - 00:1A:2B:3C:4D:5E → 192.168.1.10
5. **IP address is selected**
    - If the MAC address exists in the table, the server **selects the corresponding IP address**.

- BOOTP does **not dynamically choose any free IP** like DHCP.

6. **BOOTP reply is sent**
    - The server sends a **BOOTP Reply** to the client.
    - This reply contains:
        - Assigned IP address
        - Subnet mask
        - Default gateway
        - (Optional) Boot file name
7. **Client configures itself**
    - The client receives the reply and **configures its network interface** using the provided IP address.

**Dynamic Host Configuration Protocol (DHCP)**

The **Dynamic Host Configuration Protocol (DHCP)** is designed to provide **both static and dynamic IP address allocation**, which can be done **manually or automatically**.

---

**1. Static Address Allocation (DHCP as BOOTP)**

In **static address allocation**, DHCP works **in the same way as BOOTP**.

- DHCP is **backward compatible with BOOTP**
- A host running a **BOOTP client** can request an IP address from a **DHCP server**
- The DHCP server maintains a **static database**
- This database contains **permanent bindings** between:
    - **Physical (MAC) address**
    - **IP address**

**How it works:**

- When a client sends a request, the server looks up the client's **MAC address**
- If a match is found, the server always assigns the **same fixed IP address**

👉 This is useful for **servers, printers, and routers** that need fixed IPs.

---

**2. Dynamic Address Allocation**

In **dynamic address allocation**, DHCP assigns IP addresses **temporarily**.

- DHCP maintains a **second database (address pool)**
- This pool contains **unused IP addresses**
- When a client requests an IP address:
    1. The server selects a **free IP address** from the pool
    2. The IP is assigned for a **limited time**, called a **lease**
    3. The lease time is **negotiable** between client and server

After the lease expires:

- The client must **renew** the lease, or
- The IP address is **returned to the pool** for reuse

👉 This makes DHCP **flexible and efficient** for large networks.

Key Differences (Static vs Dynamic):

| **Feature** | **Static Allocation** | **Dynamic Allocation** |
|---|---|---|
| IP type | Fixed | Temporary |
| Database | MAC–IP binding table | Pool of free IPs |
| Lease time | No | Yes |
| Flexibility | Low | High |
| Common use | Servers, printers | User devices (PCs, phones) |

**Steps / States of DHCP**

**1. INIT State (Initializing)**

- When a DHCP client **starts for the first time**, it enters the **INIT state**.
- The client **does not have an IP address**.
- It **broadcasts a DHCPDISCOVER message** using **UDP port 67**.
- DHCPDISCOVER is a **request message** asking for IP configuration.
- Purpose: *To locate available DHCP servers on the network.*

---

**2. SELECTING State**

- After sending DHCPDISCOVER, the client enters the **SELECTING state**.
- DHCP servers that can provide service respond with **DHCPOFFER** messages.
- Each DHCPOFFER contains:
    - Offered IP address
    - Lease duration (default is usually **1 hour**)
- The server **temporarily locks** the offered IP so no other client can use it.
- The client **selects one offer** and sends a **DHCPREQUEST** to the chosen server.
- The client then moves to the **REQUESTING state**.

**Failure handling:**

- If no DHCPOFFER is received:
    - The client retries **4 more times**, every **2 seconds**
    - If still no response, it **waits 5 minutes** and tries again

---

**3. REQUESTING State**

- The client waits for confirmation from the selected server.
- The server sends a **DHCPACK** message.
- DHCPACK:
    - Confirms the IP address

    - Creates a **binding between client MAC address and IP address**
- After receiving DHCPACK, the client enters the **BOUND state**.

---

**4. BOUND State**

- The client **successfully uses the assigned IP address**.
- The IP remains valid **until the lease expires**.
- When **50% of the lease time** is reached:
    - The client sends a **DHCPREQUEST** to renew the lease
    - Moves to the **RENEWING state**
- The client may also **release the IP voluntarily** and return to INIT state.

---

**5. RENEWING State**

- The client attempts to **renew the lease with the original server**.
- Two possibilities:
    1. **DHCPACK received**
        - Lease is renewed
        - Timer resets
        - Client returns to **BOUND state**
    2. **No response**
        - When **87.5% of lease time** expires
        - Client moves to **REBINDING state**

---

**6. REBINDING State**

- The client tries to renew the lease with **any available DHCP server**.
- Three possible outcomes:
    1. **DHCPACK received**
        - Lease renewed

- Client goes to **BOUND state**

2. **DHCPNACK received**
    - Lease rejected
    - Client returns to **INIT state**
3. **Lease expires**
    - Client returns to **INIT state** and starts over

**DHCP States with Messages (Commands)**

1. **INIT** → DHCPDISCOVER
2. **SELECTING** → DHCPOFFER
3. **REQUESTING** → DHCPREQUEST
4. **BOUND** → DHCPACK
5. **RENEWING** → DHCPREQUEST
6. **REBINDING** → DHCPREQUEST
7. **RELEASE** → DHCPRELEASE

---

**Golden Memory Rule** ⭐

**DISCOVER → OFFER → REQUEST → ACK**
(DORA)

---

**Ultra-short Exam Line**

DHCP uses DHCPDISCOVER, DHCPOFFER, DHCPREQUEST, and DHCPACK messages to move through INIT, SELECTING, REQUESTING, and BOUND states, with DHCPREQUEST used again for renewing and rebinding.