

**Aim** ↔ Study basic network commands and Cisco IOS show commands.

## **Objectives** ↔

- i. Implementation of network troubleshooting command.
- ii. Utilization of show command for network analysis.

**Software Required** ↔ Cisco Packet Tracer

## **Theory** ↔

Some basic commands are critical for network diagnostics, connectivity checks, and understanding network configurations, especially in simulation environments like Cisco Packet Tracer.

- a) **ping** ↔ This command tests connectivity between your computer and another device on a network. It sends ICMP Echo Request messages and waits for Echo Reply messages, helping to determine if the other device is reachable and how long the round trip takes.
- b) **ipconfig** ↔ Displays the IP address, subnet mask, and default gateway for all network adapters in the computer. ipconfig/all provides a more detailed view, including MAC address, DHCP status, and DNS information, which help resolve human-readable domain names to IP addresses.
- c) **nslookup** ↔ Used to query DNS servers and retrieve domain name or IP address mapping information. It's useful for diagnosing DNS issues.
- d) **hostname** ↔ Returns the hostname of your computer, which is the name identified on the network.
- e) **tracert** ↔ Traces the route that packets take from your computer to a destination host. It shows each hop along the way and the time it takes for each hop, helping diagnose where delays or issues occur.
- f) **pathping** ↔ Combines the functionality of ping and tracert to provide detailed information about the route and packet loss at each hop between the source and destination. pathping google.com would give a detailed report of the route to Google's servers.
- g) **netstat** ↔ Displays network statistics, including active connections, ports on which the computer is listening, and routing table information. It's useful for monitoring network activity and troubleshooting.
- h) **getmac** ↔ Shows the MAC addresses of network adapters on the computer. MAC addresses are unique identifiers for network devices.
- i) **arp** ↔ Displays and modifies the ARP (Address Resolution Protocol) cache, which stores mappings between IP addresses and MAC addresses. It's essential for local network communications.

## Procedure ➡

- i. Drag and drop 3 PCs and 1 switch onto the workspace.
- ii. Use Copper Straight-Through cables to connect each PC to the switch.
- iii. Go to each PC's desktop, open "IP Configuration," and assign unique IP addresses within the same subnet (e.g., 8.8.8.0, 8.8.8.1, 8.8.8.2). Set the subnet mask to 255.0.0.0.
- iv. Open the command prompt on each PC and use ping to test connectivity between PCs (e.g., ping 8.8.8.1 from 8.8.8.0).
- v. Run ipconfig / ipconfig/all to check IP configurations, nslookup to test DNS lookup, hostname to verify the hostname, and tracert to trace the route between PCs or to an external address if Internet access is configured.

## Simulation ➡

```
C:\Users\nitin>ping google.com

Pinging google.com [142.250.194.174] with 32 bytes of data:
Reply from 142.250.194.174: bytes=32 time=6ms TTL=60
Reply from 142.250.194.174: bytes=32 time=9ms TTL=60
Reply from 142.250.194.174: bytes=32 time=7ms TTL=60
Reply from 142.250.194.174: bytes=32 time=7ms TTL=60

Ping statistics for 142.250.194.174:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 6ms, Maximum = 9ms, Average = 7ms

C:\Users\nitin>ipconfig

Windows IP Configuration

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::b74f:7fa5:623:2fbb%7
    IPv4 Address. . . . . : 10.10.51.15
    Subnet Mask . . . . . : 255.255.252.0
    Default Gateway . . . . . : 10.10.48.1
```

```
C:\Users\nitin>ipconfig/all
```

Windows IP Configuration

```
Host Name . . . . . : Steve
Primary Dns Suffix . . . . . :
Node Type . . . . . : Mixed
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No
```

Wireless LAN adapter Local Area Connection\* 1:

```
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter
Physical Address. . . . . : 52-5A-65-F5-C9-2F
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
```

Wireless LAN adapter Local Area Connection\* 2:

```
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #2
Physical Address. . . . . : 52-5A-65-F5-C9-3F
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
```

Wireless LAN adapter Wi-Fi:

```
Connection-specific DNS Suffix . :
Description . . . . . : MediaTek Wi-Fi 6 MT7921 Wireless LAN Card
Physical Address. . . . . : 50-5A-65-F5-C9-6F
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::b74f:7fa5:623:2fbb%7(Preferred)
IPv4 Address. . . . . : 10.10.51.15(Preferred)
Subnet Mask . . . . . : 255.255.252.0
Lease Obtained. . . . . : 15 August 2024 01:14:49
Lease Expires . . . . . : 15 August 2024 10:14:50
Default Gateway . . . . . : 10.10.48.1
DHCP Server . . . . . : 1.1.1.1
DHCPv6 IAID . . . . . : 105929317
DHCPv6 Client DUID. . . . . : 00-01-00-01-2B-F8-67-7D-00-DE-AB-CA-60-4F
DNS Servers . . . . . : 8.8.8.8
                        8.8.4.4
                        4.2.2.2
NetBIOS over Tcpip. . . . . : Enabled
```

```
C:\Users\nitin>nslookup
Default Server:  dns.google
Address:  8.8.8.8

> exit

C:\Users\nitin>hostname
Steve

C:\Users\nitin>tracert

Usage: tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout]
              [-R] [-S srcaddr] [-4] [-6] target_name

Options:
    -d                      Do not resolve addresses to hostnames.
    -h maximum_hops        Maximum number of hops to search for target.
    -j host-list            Loose source route along host-list (IPv4-only).
    -w timeout              Wait timeout milliseconds for each reply.
    -R                      Trace round-trip path (IPv6-only).
    -S srcaddr              Source address to use (IPv6-only).
    -4                      Force using IPv4.
    -6                      Force using IPv6.

C:\Users\nitin>pathping

Usage: pathping [-g host-list] [-h maximum_hops] [-i address] [-n]
               [-p period] [-q num_queries] [-w timeout]
               [-4] [-6] target_name

Options:
    -g host-list            Loose source route along host-list.
    -h maximum_hops        Maximum number of hops to search for target.
    -i address              Use the specified source address.
    -n                      Do not resolve addresses to hostnames.
    -p period               Wait period milliseconds between pings.
    -q num_queries          Number of queries per hop.
    -w timeout              Wait timeout milliseconds for each reply.
    -4                      Force using IPv4.
    -6                      Force using IPv6.
```

```
C:\Users\nitin>pathping google.com
```

```
Tracing route to google.com [142.250.194.174]  
over a maximum of 30 hops:
```

```
0 Steve [10.10.51.15]
```

```
1 10.10.48.1
```

```
2 * * *
```

```
Computing statistics for 25 seconds...
```

Hop	RTT	Source to Here Lost/Sent = Pct	This Node/Link Lost/Sent = Pct	Address
0				Steve [10.10.51.15]
1	32ms	1/ 100 = 1%	1/ 100 = 1%	
			0/ 100 = 0%	10.10.48.1

```
Trace complete.
```

```
C:\Users\nitin>netstat
```

```
Active Connections
```

Proto	Local Address	Foreign Address	State
TCP	10.10.51.15:49408	20.198.119.143:https	ESTABLISHED
TCP	10.10.51.15:49868	20.198.118.190:https	ESTABLISHED
TCP	10.10.51.15:49916	20.167.82.225:https	ESTABLISHED
TCP	10.10.51.15:49980	ec2-44-218-192-198:https	ESTABLISHED
TCP	10.10.51.15:49996	40.99.31.162:https	TIME_WAIT
TCP	10.10.51.15:50118	dns:https	TIME_WAIT
TCP	10.10.51.15:50119	server-18-239-142-21:https	TIME_WAIT
TCP	10.10.51.15:50120	server-54-182-0-51:https	TIME_WAIT
TCP	10.10.51.15:50121	server-18-239-142-21:https	TIME_WAIT
TCP	10.10.51.15:50123	sc-in-f84:https	TIME_WAIT
TCP	10.10.51.15:50124	server-18-172-64-31:https	TIME_WAIT
TCP	10.10.51.15:50126	47:https	TIME_WAIT
TCP	10.10.51.15:50127	server-18-239-142-44:https	TIME_WAIT
TCP	10.10.51.15:50128	server-18-239-142-44:https	TIME_WAIT
TCP	10.10.51.15:50129	server-108-159-80-47:https	TIME_WAIT
TCP	10.10.51.15:50133	del11s22-in-f14:https	TIME_WAIT
TCP	10.10.51.15:50134	sin26s10-in-f6:https	TIME_WAIT
TCP	10.10.51.15:50136	ec2-52-5-62-219:https	ESTABLISHED
TCP	10.10.51.15:50140	104.18.43.79:https	TIME_WAIT
TCP	10.10.51.15:50142	104.18.40.222:https	TIME_WAIT
TCP	10.10.51.15:50147	kul01s10-in-f42:https	TIME_WAIT
TCP	10.10.51.15:50148	172.64.146.223:https	TIME_WAIT
TCP	10.10.51.15:50154	104.18.86.42:https	TIME_WAIT
TCP	10.10.51.15:50155	104.18.86.42:https	TIME_WAIT
TCP	10.10.51.15:50156	sc-in-f84:https	TIME_WAIT
TCP	10.10.51.15:50157	104.18.28.127:https	TIME_WAIT
TCP	10.10.51.15:50158	104.18.35.23:https	TIME_WAIT
TCP	10.10.51.15:50159	ec2-52-11-247-82:https	ESTABLISHED
TCP	10.10.51.15:50160	a23-54-81-209:https	ESTABLISHED
TCP	10.10.51.15:50161	52.109.124.28:https	TIME_WAIT
TCP	10.10.51.15:50162	52.168.112.66:https	TIME_WAIT
TCP	10.10.51.15:50164	a23-57-205-123:http	ESTABLISHED
TCP	10.10.51.15:50165	a23-57-205-123:https	ESTABLISHED
TCP	10.10.51.15:50166	a23-57-205-123:http	ESTABLISHED
TCP	10.10.51.15:50167	a23-212-160-85:http	TIME_WAIT
TCP	10.10.51.15:50169	a23-212-160-85:https	ESTABLISHED
TCP	10.10.51.15:50170	a23-57-205-123:https	ESTABLISHED
TCP	10.10.51.15:50171	server-18-239-142-117:https	ESTABLISHED
TCP	10.10.51.15:50172	a23-57-207-82:https	ESTABLISHED
TCP	10.10.51.15:50175	40.99.31.162:https	ESTABLISHED
TCP	10.10.51.15:50176	a23-54-83-203:https	ESTABLISHED
TCP	10.10.51.15:50179	20.42.73.24:https	ESTABLISHED
TCP	10.10.51.15:50180	a104-90-6-226:http	TIME_WAIT
TCP	10.10.51.15:50181	150.171.28.254:https	ESTABLISHED
TCP	10.10.51.15:50182	13.107.4.254:https	ESTABLISHED

```
C:\Users\nitin>getmac
```

Physical Address	Transport Name
50-5A-65-F5-C9-6F	\Device\NPF{551A0738-AB90-4FDC-B72C-774F8D298BFC}

```
C:\Users\nitin>arp
```

Displays and modifies the IP-to-Physical address translation tables used by address resolution protocol (ARP).

```
ARP -s inet_addr eth_addr [if_addr]
```

```
ARP -d inet_addr [if_addr]
```

```
ARP -a [inet_addr] [-N if_addr] [-v]
```

-a Displays current ARP entries by interrogating the current protocol data. If inet\_addr is specified, the IP and Physical addresses for only the specified computer are displayed. If more than one network interface uses ARP, entries for each ARP table are displayed.

-g Same as -a.

-v Displays current ARP entries in verbose mode. All invalid entries and entries on the loop-back interface will be shown.

inet\_addr Specifies an internet address.

-N if\_addr Displays the ARP entries for the network interface specified by if\_addr.

-d Deletes the host specified by inet\_addr. inet\_addr may be wildcarded with \* to delete all hosts.

-s Adds the host and associates the Internet address inet\_addr with the Physical address eth\_addr. The Physical address is given as 6 hexadecimal bytes separated by hyphens. The entry is permanent.

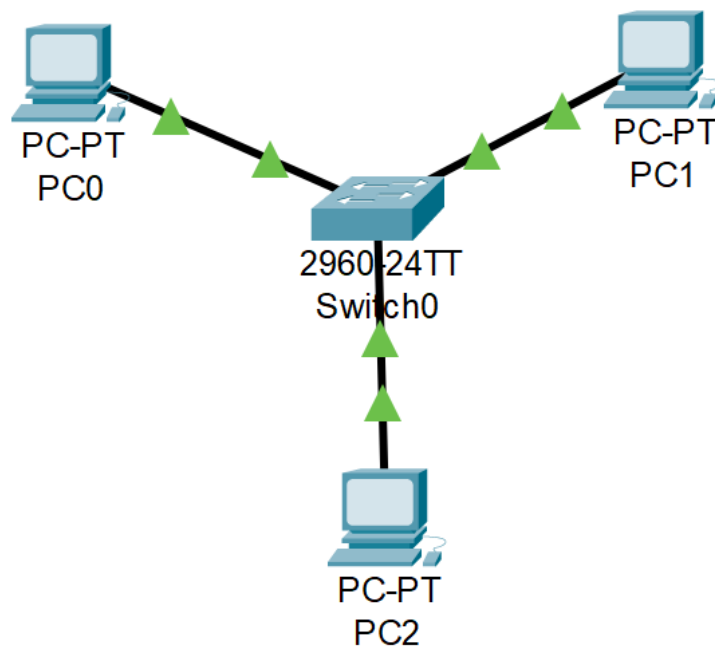
eth\_addr Specifies a physical address.

if\_addr If present, this specifies the Internet address of the interface whose address translation table should be modified. If not present, the first applicable interface will be used.

Example:

```
> arp -s 157.55.85.212 00-aa-00-62-c6-09 .... Adds a static entry.
```

```
> arp -a .... Displays the arp table.
```



## **Result ↗**

In this experiment, we successfully created a network with three PCs connected through a switch in Cisco Packet Tracer. Each PC was assigned a unique IP address within the same subnet. Connectivity was verified using commands like ping, ipconfig, tracert, and pathping, all of which confirmed successful communication and proper configuration.

## **Conclusion ↗**

This experiment demonstrated basic network setup and verification using essential commands in Cisco Packet Tracer. Effective communication between PCs was established, providing a foundation for further network studies.

## **Precautions ↗**

- Assign unique IP addresses within the same subnet to each PC.
- Use the correct Copper Straight-Through cables for connections.
- Ensure all PCs have the correct subnet mask.
- Carefully execute each command and monitor the results for accuracy.