

# Topic 9: Network Implementation Essentials

Introduction to Networks v7.0  
(ITN)



# Devices in a Small Network

# Small Network Topologies

- The majority of businesses are small most of the business networks are also small.
- A small network design is usually simple.
- Small networks typically have a single WAN connection provided by DSL, cable, or an Ethernet connection.
- Large networks require an IT department to maintain, secure, and troubleshoot network devices and to protect organizational data. Small networks are managed by a local IT technician or by a contracted professional.

# Device Selection for a Small Network

Like large networks, small networks require planning and design to meet user requirements. Planning ensures that all requirements, cost factors, and deployment options are given due consideration. One of the first design considerations is the type of intermediary devices to use to support the network.

Factors that must be considered when selecting network devices include:

- cost
- speed and types of ports/interfaces
- expandability
- operating system features and services

# IP Addressing for a Small Network

When implementing a network, create an IP addressing scheme and use it. All hosts and devices within an internetwork must have a unique address. Devices that will factor into the IP addressing scheme include the following:

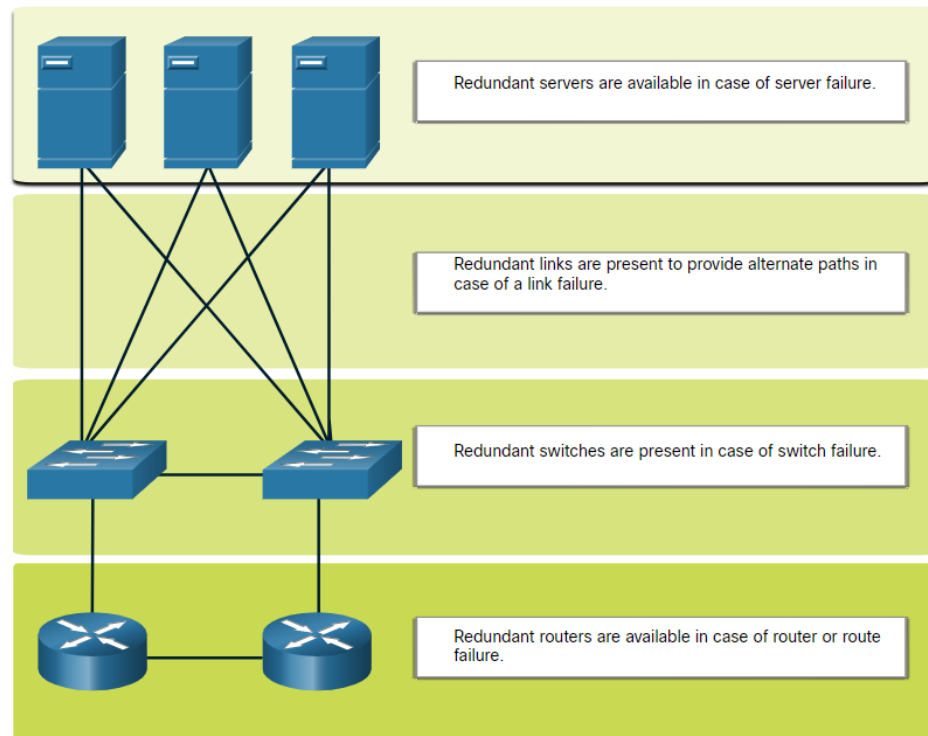
- End user devices - The number and type of connections (i.e., wired, wireless, remote access)
- Servers and peripherals devices (e.g., printers and security cameras)
- Intermediary devices including switches and access points

It is recommended that you plan, document, and maintain an IP addressing scheme based on device type. The use of a planned IP addressing scheme makes it easier to identify a type of device and to troubleshoot problems.

# Redundancy in a Small Network

In order to maintain a high degree of reliability, *redundancy* is required in the network design. Redundancy helps to eliminate single points of failure.

Redundancy can be accomplished by installing duplicate equipment. It can also be accomplished by supplying duplicate network links for critical areas.

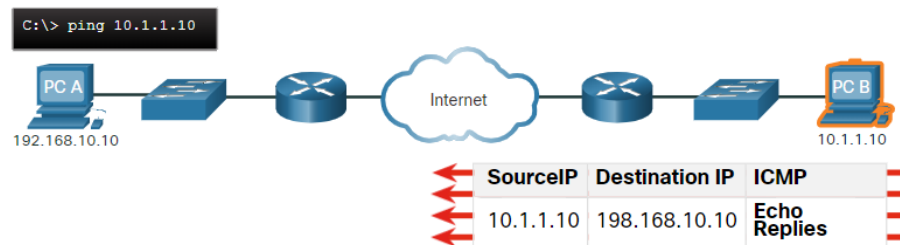
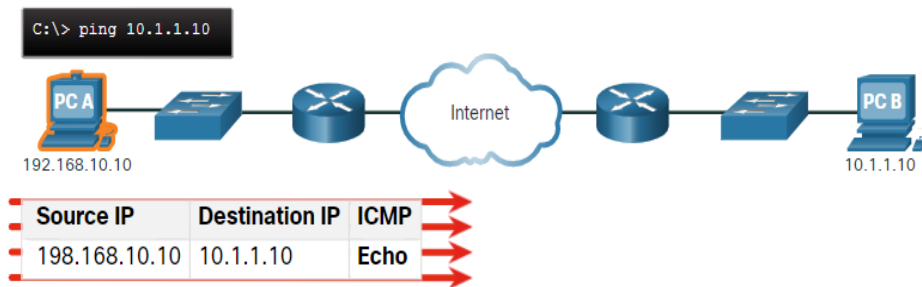


# Verify Connectivity

# Verify Connectivity with Ping

Whether your network is small and new, or you are scaling an existing network, you will always want to be able to verify that your components are properly connected to each other and to the internet.

- The ping command, available on most operating systems, is the most effective way to quickly test Layer 3 connectivity between a source and destination IP address.
- The ping command uses the Internet Control Message Protocol (ICMP) echo (ICMP Type 8) and echo reply (ICMP Type 0) messages.





# Verify Connectivity with Ping (Cont.)

On a Windows 10 host, the ping command sends four consecutive ICMP echo messages and expects four consecutive ICMP echo replies from the destination. The IOS ping sends five ICMP echo messages and displays an indicator for each ICMP echo reply received.

IOS Ping Indicators are as follows:

Element	Description
!	<ul style="list-style-type: none"><li>•Exclamation mark indicates successful receipt of an echo reply message.</li><li>•It validates a Layer 3 connection between source and destination.</li></ul>
.	<ul style="list-style-type: none"><li>•A period means that time expired waiting for an echo reply message.</li><li>•This indicates a connectivity problem occurred somewhere along the path.</li></ul>
U	<ul style="list-style-type: none"><li>•Uppercase <b>U</b> indicates a router along the path responded with an ICMP Type 3 “destination unreachable” error message.</li><li>•Possible reasons include the router does not know the direction to the destination network or it could not find the host on the destination network.</li></ul>

**Note:** Other possible ping replies include Q, M, ?, or &. However, the meaning of these are out of scope for this module.

# Verify Connectivity Extended Ping

The Cisco IOS offers an "extended" mode of the **ping** command.

Extended ping is entered in privileged EXEC mode by typing **ping** without a destination IP address. You will then be given several prompts to customize the extended **ping**.

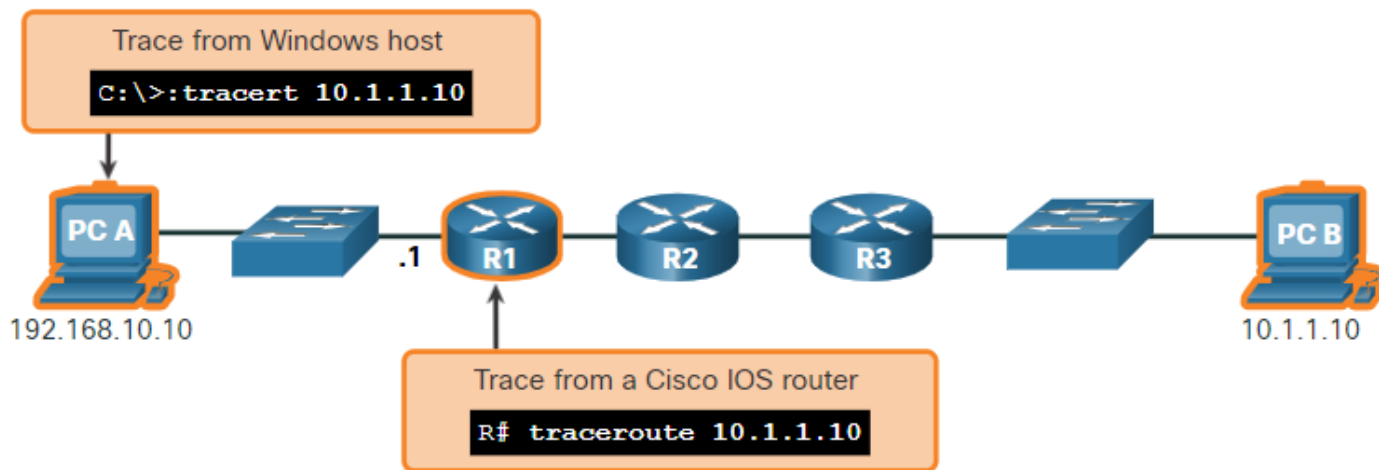
**Note:** Pressing **Enter** accepts the indicated default values. The **ping ipv6** command is used for IPv6 extended pings.

```
R1# ping
Protocol [ip]:
Target IP address: 10.1.1.10
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Ingress ping [n]:
Source address or interface: 192.168.10.1
DSCP Value [0]:
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0x0000ABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.10.1
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R1#
```

# Verify Connectivity with Traceroute

The ping command is useful to quickly determine if there is a Layer 3 connectivity problem. However, it does not identify where the problem is located along the path.

- Traceroute can help locate Layer 3 problem areas in a network. A trace returns a list of hops as a packet is routed through a network.
- The syntax of the trace command varies between operating systems.



# Verify Connectivity with Traceroute (Cont.)

- The following is a sample output of **tracert** command on a Windows 10 host.

**Note:** Use **Ctrl-C** to interrupt a **tracert** in Windows.

- The only successful response was from the gateway on R1. Trace requests to the next hop timed out as indicated by the asterisk (\*), meaning that the next hop router did not respond or there is a failure in the network path. In this example there appears to be a problem between R1 and R2.

```
C:\Users\PC-A> tracert 10.1.1.10
Tracing route to 10.1.10 over a maximum of 30 hops:
  1      2 ms      2 ms      2 ms    192.168.10.1
  2      *        *        *      Request timed out.
  3      *        *        *      Request timed out.
  4      *        *        *      Request timed out.
^C
C:\Users\PC-A>
```

# Verify Connectivity with Traceroute (Cont.)

The following are sample outputs of traceroute command from R1:

```
R1# traceroute 10.1.1.10
Type escape sequence to abort.
Tracing the route to 10.1.1.10
VRF info: (vrf in name/id, vrf out name/id)
 1 209.165.200.226 1 msec 0 msec 1 msec
 2 209.165.200.230 1 msec 0 msec 1 msec
 3 10.1.1.10 1 msec 0 msec
R1#
```

```
R1# traceroute 10.1.1.10
Type escape sequence to abort.
Tracing the route to 10.1.1.10
VRF info: (vrf in name/id, vrf out name/id)
 1 209.165.200.226 1 msec 0 msec 1 msec
 2 209.165.200.230 1 msec 0 msec 1 msec
 3 * * *
 4 * * *
 5 *
```

- On the left, the trace validated that it could successfully reach PC B.
- On the right, the 10.1.1.10 host was not available, and the output shows asterisks where replies timed out. Timeouts indicate a potential network problem.
- Use **Ctrl-Shift-6** to interrupt a **traceroute** in Cisco IOS.

**Note:** Windows implementation of traceroute (tracert) sends ICMP Echo Requests. Cisco IOS and Linux use UDP with an invalid port number. The final destination will return an ICMP port unreachable message.

# Verify Connectivity

## Extended Traceroute

Like the extended **ping** command, there is also an extended **tracert** command. It allows the administrator to adjust parameters related to the command operation.

The Windows **tracert** command allows the input of several parameters through options in the command line. However, it is not guided like the extended traceroute IOS command. The following output displays the available options for the Windows **tracert** command:

```
C:\Users\PC-A> 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\PC-A>
```

# Extended Traceroute (Cont.)

- The Cisco IOS extended **traceroute** option enables the user to create a special type of trace by adjusting parameters related to the command operation.
- Extended traceroute is entered in privileged EXEC mode by typing **traceroute** without a destination IP address. IOS will guide you through the command options by presenting a number of prompts related to the setting of all the different parameters.
- **Note:** Pressing **Enter** accepts the indicated default values.

```
R1# traceroute
Protocol [ip]:
Target IP address: 10.1.1.10
Ingress traceroute [n]:
Source address: 192.168.10.1
DSCP Value [0]:
Numeric display [n]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Port Number [33434]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Type escape sequence to abort.
Tracing the route to 192.168.10.10
VRF info: (vrf in name/id, vrf out name/id)
  1 209.165.200.226 1 msec 1 msec 1 msec
  2 209.165.200.230 0 msec 1 msec 0 msec
  3 *
    10.1.1.10 2 msec 2 msec
R1#
```

# Verify Connectivity

## Network Baseline

- One of the most effective tools for monitoring and troubleshooting network performance is to establish a network baseline.
- One method for starting a baseline is to copy and paste the results from an executed ping, trace, or other relevant commands into a text file. These text files can be time stamped with the date and saved into an archive for later retrieval and comparison.
- Among items to consider are error messages and the response times from host to host.
- Corporate networks should have extensive baselines; more extensive than we can describe in this course. Professional-grade software tools are available for storing and maintaining baseline information.



# Host and IOS Commands

# IP Configuration on a Windows Host

In Windows 10, you can access the IP address details from the **Network and Sharing Center** to quickly view the four important settings: address, mask, router, and DNS. Or you can issue the **ipconfig** command at the command line of a Windows computer.

- Use the **ipconfig /all** command to view the MAC address, as well as a number of details regarding the Layer 3 addressing of the device.
- If a host is configured as a DHCP client, the IP address configuration can be renewed using the **ipconfig /release** and **ipconfig /renew** commands.
- The DNS Client service on Windows PCs also optimizes the performance of DNS name resolution by storing previously resolved names in memory. The **ipconfig /displaydns** command displays all of the cached DNS entries on a Windows computer system.

```
C:\Users\PC-A> ipconfig
Windows IP Configuration
(Output omitted)
Wireless LAN adapter Wi-Fi:
    Connection-specific DNS Suffix . : 
    Link-local IPv6 Address . . . . . : fe80::a4aa:2dd1:ae2d:a75e%16
    IPv4 Address. . . . . : 192.168.10.10
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.10.1
(Output omitted)
```

# Host and IOS Commands

## The arp Command

The **arp** command is executed from the Windows, Linux, or Mac command prompt. The command lists all devices currently in the ARP cache of the host.

- The **arp -a** command displays the known IP address and MAC address binding. The ARP cache only displays information from devices that have been recently accessed.
- To ensure that the ARP cache is populated, **ping** a device so that it will have an entry in the ARP table.
- The cache can be cleared by using the **netsh interface ip delete arpcache** command in the event the network administrator wants to repopulate the cache with updated information.

**Note:** You may need administrator access on the host to be able to use the **netsh interface ip delete arpcache** command.

# Common show Commands Revisited

Command	Description
show running-config	Verifies the current configuration and settings
show interfaces	Verifies the interface status and displays any error messages
show ip interface	Verifies the Layer 3 information of an interface
show arp	Verifies the list of known hosts on the local Ethernet LANs
show ip route	Verifies the Layer 3 routing information
show protocols	Verifies which protocols are operational
show version	Verifies the memory, interfaces, and licenses of the device

# The show cdp neighbors Command

CDP provides the following information about each CDP neighbor device:

- **Device identifiers** - The configured host name of a switch, router, or other device
- **Address list** - Up to one network layer address for each protocol supported
- **Port identifier** - The name of the local and remote port in the form of an ASCII character string, such as FastEthernet 0/0
- **Capabilities list** - Whether a specific device is a Layer 2 switch or a Layer 3 switch
- **Platform** - The hardware platform of the device.

The **show cdp neighbors detail** command reveals the IP address of a neighboring device.

```
R3# show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID         Local Intrfce   Holdtme    Capability   Platform   Port ID
S3                 Gig 0/0/1       122        S I         WS-C2960+  Fas 0/5
Total cdp entries displayed : 1
R3#
```

# The show ip interface brief Command

One of the most frequently used commands is the **show ip interface brief** command. This command provides a more abbreviated output than the **show ip interface** command. It provides a summary of the key information for all the network interfaces on a router.

```
R1# show ip interface brief
```

Interface	IP-Address	OK?	Method	Status	Protocol
GigabitEthernet0/0/0	209.165.200.225	YES	manual	up	up
GigabitEthernet0/0/1	192.168.10.1	YES	manual	up	up
Serial0/1/0	unassigned	NO	unset	down	down
Serial0/1/1	unassigned	NO	unset	down	down
GigabitEthernet0	unassigned	YES	unset	administratively down	down

```
R1#
```

```
S1# show ip interface brief
```

Interface	IP-Address	OK?	Method	Status	Protocol
Vlan1	192.168.254.250	YES	manual	up	up
FastEthernet0/1	unassigned	YES	unset	down	down
FastEthernet0/2	unassigned	YES	unset	up	up
FastEthernet0/3	unassigned	YES	unset	up	up

# Troubleshooting Methodologies

# Troubleshooting Methodologies

## Basic Troubleshooting Approaches

Step	Description
<b>Step 1. Identify the Problem</b>	<ul style="list-style-type: none"><li>• This is the first step in the troubleshooting process.</li><li>• Although tools can be used in this step, a conversation with the user is often very helpful.</li></ul>
<b>Step 2. Establish a Theory of Probable Causes</b>	<ul style="list-style-type: none"><li>• After the problem is identified, try to establish a theory of probable causes.</li><li>• This step often yields more than a few probable causes to the problem.</li></ul>
<b>Step 3. Test the Theory to Determine Cause</b>	<ul style="list-style-type: none"><li>• Based on the probable causes, test your theories to determine which one is the cause of the problem.</li><li>• A technician may apply a quick fix to test and see if it solves the problem.</li><li>• If a quick fix does not correct the problem, you might need to research the problem further to establish the exact cause.</li></ul>
<b>Step 4. Establish a Plan of Action and Implement the Solution</b>	After you have determined the exact cause of the problem, establish a plan of action to resolve the problem and implement the solution.
<b>Step 5. Verify Solution and Implement Preventive Measures</b>	<ul style="list-style-type: none"><li>• After you have corrected the problem, verify full functionality.</li><li>• If applicable, implement preventive measures.</li></ul>
<b>Step 6. Document Findings, Actions, and Outcomes</b>	<ul style="list-style-type: none"><li>• In the final step of the troubleshooting process, document your findings, actions, and outcomes.</li><li>• This is very important for future reference.</li></ul>



# Troubleshooting Methodologies

## Resolve or Escalate?

- In some situations, it may not be possible to resolve the problem immediately. A problem should be escalated when it requires a manager decision, some specific expertise, or network access level unavailable to the troubleshooting technician.
- A company policy should clearly state when and how a technician should escalate a problem.

# Troubleshooting Methodologies

## The debug Command

- The IOS **debug** command allows the administrator to display OS process, protocol, mechanism and event messages in real-time for analysis.
- All **debug** commands are entered in privileged EXEC mode. The Cisco IOS allows for narrowing the output of **debug** to include only the relevant feature or subfeature. Use **debug** commands only to troubleshoot specific problems.
- To list a brief description of all the debugging command options, use the **debug ?** command in privileged EXEC mode at the command line.
- To turn off a specific debugging feature, add the **no** keyword in front of the **debug** command
- Alternatively, you can enter the **undebug** form of the command in privileged EXEC mode.
- To turn off all active debug commands at once, use the **undebug all** command.
- Be cautious using some **debug** commands, as they may generate a substantial amount of output and use a large portion of system resources. The router could get so busy displaying **debug** messages that it would not have enough processing power to perform its network functions, or even listen to commands to turn off debugging.

# Troubleshooting Methodologies

## The terminal monitor Command

- **debug** and certain other IOS message output is not automatically displayed on remote connections. This is because log messages are prevented from being displayed on vty lines.
- To display log messages on a terminal (virtual console), use the **terminal monitor** privileged EXEC command. To stop logging messages on a terminal, use the **terminal no monitor** privileged EXEC command.

```
R2# telnet 209.165.200.225
Trying 209.165.200.225 ... Open
  Authorized access only!
  User Access Verification
Password:
R1> enable
Password:
R1# debug ip icmp
ICMP packet debugging is on
R1# ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
R1#
```

```
R1# terminal monitor
R1# ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
R1#
*Aug 20 16:03:49.735: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225, topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.737: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225, topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.738: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225, topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.740: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225, topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.741: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225, topology BASE, dscp 0
topoid 0
R1# no debug ip icmp
ICMP packet debugging is off
R1#
```

# Troubleshooting Scenarios

# Duplex Operation and Mismatch Issues

- Interconnecting Ethernet interfaces must operate in the same duplex mode for best communication performance and to avoid inefficiency and latency on the link.
- The Ethernet autonegotiation feature facilitates configuration, minimizes problems and maximizes link performance between two interconnecting Ethernet links. The connected devices first announce their supported capabilities and then choose the highest performance mode supported by both ends.
- If one of the two connected devices is operating in full-duplex and the other is operating in half-duplex, a duplex mismatch occurs. While data communication will occur through a link with a duplex mismatch, link performance will be very poor.
- Duplex mismatches are typically caused by a misconfigured interface or in rare instances by a failed autonegotiation. Duplex mismatches may be difficult to troubleshoot as the communication between devices still occurs.

# IP Addressing Issues on IOS Devices

- Two common causes of incorrect IPv4 assignment are manual assignment mistakes or DHCP-related issues.
- Network administrators often have to manually assign IP addresses to devices such as servers and routers. If a mistake is made during the assignment, then communications issues with the device are very likely to occur.
- On an IOS device, use the **show ip interface** or **show ip interface brief** commands to verify what IPv4 addresses are assigned to the network interfaces. For example, issuing the **show ip interface brief** command as shown would validate the interface status on R1.

```
R1# show ip interface brief
```

Interface	IP-Address	OK?	Method	Status	Protocol
GigabitEthernet0/0/0	209.165.200.225	YES	manual	up	up
GigabitEthernet0/0/1	192.168.10.1	YES	manual	up	up
Serial0/1/0	unassigned	NO	unset	down	down
Serial0/1/1	unassigned	NO	unset	down	down
GigabitEthernet0	unassigned	YES	unset	administratively down	down

```
R1#
```

# IP Addressing Issues on End Devices

- On Windows-based machines, when the device cannot contact a DHCP server, Windows will automatically assign an address belonging to the 169.254.0.0/16 range. This feature is called Automatic Private IP Addressing (APIPA).
- A computer with an APIPA address will not be able to communicate with other devices in the network because those devices will most likely not belong to the 169.254.0.0/16 network.
- **Note:** Other operating systems, such as Linux and OS X, do not use APIPA.
- If the device is unable to communicate with the DHCP server, then the server cannot assign an IPv4 address for the specific network and the device will not be able to communicate.
- To verify the IP addresses assigned to a Windows-based computer, use the **ipconfig** command.

## Troubleshooting Scenarios

# Default Gateway Issues

- The default gateway for an end device is the closest networking device, belonging to the same network as the end device, that can forward traffic to other networks. If a device has an incorrect or nonexistent default gateway address, it will not be able to communicate with devices in remote networks.
- Similar to IPv4 addressing issues, default gateway problems can be related to misconfiguration (in the case of manual assignment) or DHCP problems (if automatic assignment is in use).
- To verify the default gateway on Windows-based computers, use the **ipconfig** command.
- On a router, use the **show ip route** command to list the routing table and verify that the default gateway, known as a default route, has been set. This route is used when the destination address of the packet does not match any other routes in its routing table.



## Troubleshooting Scenarios

# Troubleshooting DNS Issues

- It is common for users to mistakenly relate the operation of an internet link to the availability of the DNS.
- DNS server addresses can be manually or automatically assigned via DHCP.
- Although it is common for companies and organizations to manage their own DNS servers, any reachable DNS server can be used to resolve names.
- Cisco offers OpenDNS which provides secure DNS service by filtering phishing and some malware sites. OpenDNS addresses are 208.67.222.222 and 208.67.220.220. Advanced features such as web content filtering and security are available to families and businesses.
- Use the **ipconfig /all** as shown to verify which DNS server is in use by the Windows computer.
- The **nslookup** command is another useful DNS troubleshooting tool for PCs. With **nslookup** a user can manually place DNS queries and analyze the DNS response.