COURSE: **NETWORK ADMINISTRATION AND MANAGEMENT**

COURSE INSTRUCTOR: **MIRALDA CUKA, PHD**

Lecture 6
Dynamic Routing

Dynamic Routing

- Dynamic routing less time-consuming than static routes
- Implemented in any type of network consisting of more than just a few routers.
- Dynamic routing protocols are scalable and automatically determine better routes if there is a change in the topology.

Dynamic routing protocols are commonly used in the following scenarios:

- In networks consisting of more than just a few routers
- When a change in the network topology requires the network to automatically determine another path
- As network grows, the dynamic routing protocol automatically learns about any new networks.

Dynamic Routing Protocols

- One of the first routing protocols was RIP.
- As networks evolved and became more complex, new routing protocols emerged.
- The RIP protocol was updated to RIPv2 to accommodate growth in the network environment but it still does not scale to the larger network implementations.
- To address the needs of larger networks, two advanced routing protocols were developed:
  - Open Shortest Path First (OSPF).
  - Enhanced IGRP (EIGRP)
- Additionally, there was the need to connect the different routing domains of different organizations and provide routing between them.
  - The Border Gateway Protocol (BGP).
  - BGP is also used between ISPs and some private organizations to exchange routing information.

Dynamic Routing Protocols (cont.)

Distance Vector Routing Protocol

In this routing scheme, each router periodically shares its knowledge about the entire network with its neighbors. Each router has a table with information about network. These tables are updated by exchanging information with the immediate neighbors.

Link State Routing Protocol

In the case of Link state routing protocol, you can send the full information once in the network and if there is some update and changes next time, you are going to send only the changes across the network and that is way of working of the Link State Routing Protocol. Link-State router tells ALL other routers about ONLY its neighbors and links.

Dynamic Routing Protocols (cont.)

Features	Distance Vector	Link State
Convergence	Slow	Fast
Updates	Frequently	Event Triggered
Loops	Prone to routing Loops	Less Subjected to Routing Loops
Configuration	Easy	Difficult
Network Types	Broadcast for updates sent	Multicast for updates sent
Topology	doesn't know Network Topology	Knows entire Network Topology
Automatic Route Summarization	No	Yes
Path Calculation	Hop Count	Shortest Path -Metric
Scalability	Limited	Can be highly scalable
Protocols	RIP, IGRP	OSPF, IS-IS
Algorithm	Bredford Algorithm	Dijkstra-algorithm
Manual Route Summarization	Yes	Yes
Metric	Hop Count	Link Cost

Static vs Dynamic Routes

Feature	Dynamic Routing	Static Routing
Configuration complexity	Independent of network size	Increases with network size
Topology changes	Automatically adapts to topology changes	Administrator intervention required
Scalability	Suitable for simple to complex network topologies	Suitable for simple topologies
Security	Security must be configured	Security is inherent
Resource Usage	Uses CPU, memory, and link bandwidth	No additional resources needed
Path Predictability	Route depends on topology and routing protocol used	Explicitly defined by the administrator

Purpose of Dynamic Routing Protocols

The purpose of dynamic routing protocols includes the following:

- Discovery of remote networks
- Maintaining up-to-date routing information
- Choosing the best path to destination networks
- Ability to find a new best path if the current path is no longer available

Purpose of Dynamic Routing Protocols (cont.)

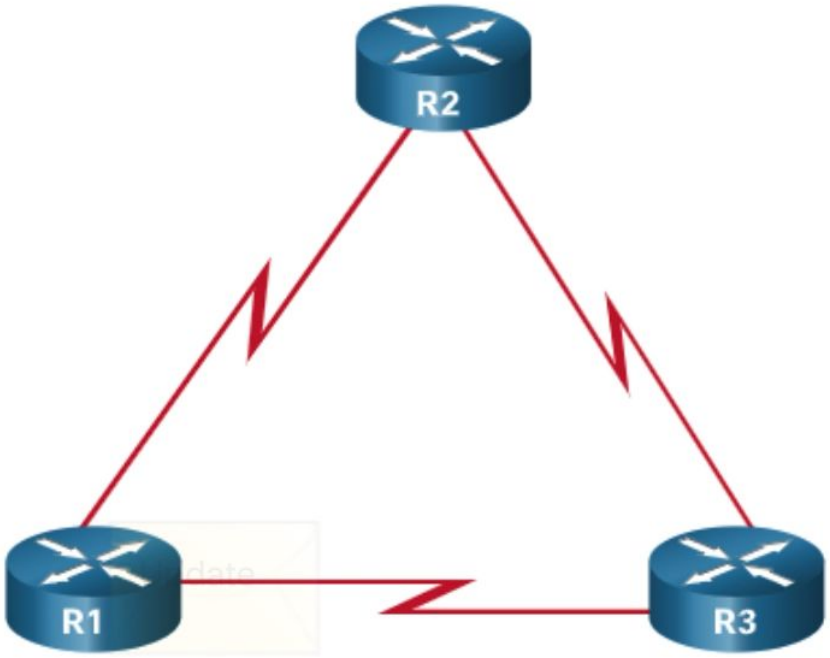
The main components of dynamic routing protocols include the following:

- **Data structures** - Routing protocols typically use tables or databases for their operations. This information is kept in RAM.
- **Routing protocol messages** - Routing protocols use various types of messages to *discover neighboring routers*, *exchange routing information*, and other tasks to learn and maintain accurate information about the network.
- **Algorithm** - An algorithm is a finite list of steps used to accomplish a task. Routing protocols use algorithms for facilitating routing information and for the best path determination.

Purpose of Dynamic Routing Protocols (cont.)

- Routing protocols determine the best path, or route, to each network.
- The route will be installed in the routing table if there is not another routing source with a lower AD.

Routing protocols allow routers to dynamically share information about remote networks and automatically offer this information to their own routing tables.



- A primary benefit of dynamic routing protocols is that routers exchange routing information when there is a topology change.
- This exchange allows routers to automatically learn about new networks and to find alternate paths when there is a link failure to a current network.

Best Path

- Determining the best path involves the evaluation of multiple paths to the same destination network and selecting the shortest path to reach that network.
- When multiple paths to the same network exist, each path uses a different exit interface on the router to reach that network.
- The best path is selected by a routing protocol based on the value or metric it uses to determine the distance to reach a network.
- A metric is the quantitative value used to measure the distance to a given network.
- The best path to a network is the path with the lowest metric.

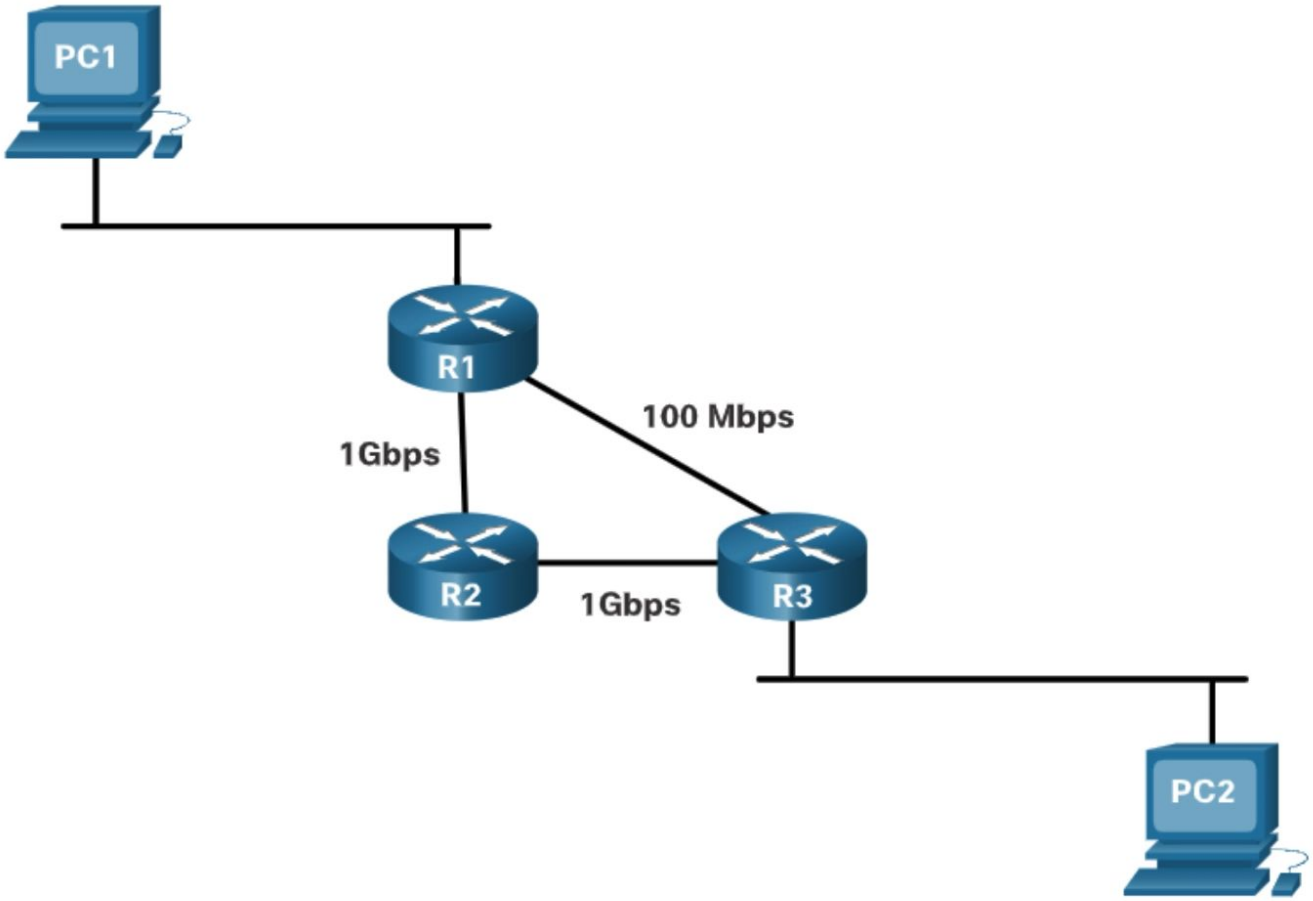
Best Path (cont.)

- Dynamic routing protocols typically use their own rules and metrics to build and update routing tables.
- The routing algorithm generates a value, or a metric, for each path through the network.
- Metrics can be based on either a single characteristic or several characteristics of a path.
- Some routing protocols can base route selection on multiple metrics, combining them into a single metric.
- The following table lists common dynamic protocols and their metrics.

Routing Protocol	Metric
Routing Information Protocol (RIP)	• The metric is "hop count". • Each router along a path adds a hop to the hop count. • A maximum of 15 hops allowed.
Open Shortest Path First (OSPF)	• The metric is "cost" which is the based on the cumulative bandwidth from source to destination. • Faster links are assigned lower costs compared to slower (higher cost) links.
Enhanced Interior Gateway Routing Protocol (EIGRP)	• It calculates a metric based on the slowest bandwidth and delay values. • It could also include load and reliability into the metric calculation.

Best Path Determined by Different Metrics

- The path may be different depending on the metric being used.
- If the best path fails, the dynamic routing protocol will automatically select a new best path if one exists.

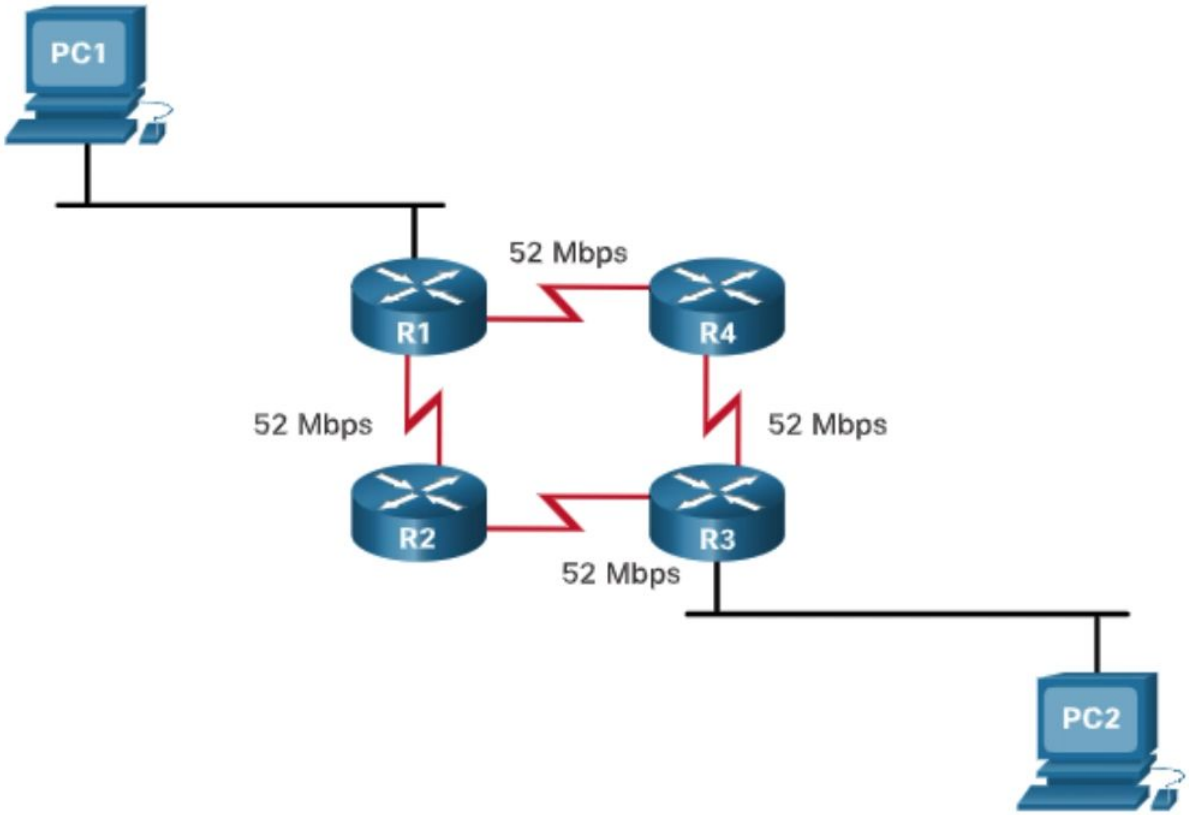


Load Balancing

- When a router has two or more paths to a destination with equal cost metrics, then the router forwards the packets using both paths equally.
- This is called equal cost load balancing.
- The routing table contains the single destination network, but has multiple exit interfaces, one for each equal cost path.
- The router forwards packets using the multiple exit interfaces listed in the routing table.
- Load balancing can increase the effectiveness and performance of the network.
- Equal cost load balancing is implemented automatically by dynamic routing protocols.
- Note: Only EIGRP supports unequal cost load balancing.

Load Balancing

- Equal cost load balancing



Advantages and Disadvantages of Dynamic Routing

- Advantages of dynamic routing include:
  - Automatically share information about remote networks
  - Determine the best path to each network and add this information to their routing tables.
  - Compared to static routing, dynamic routing protocols require less administrative overhead.
  - Help the network administrator manage the time-consuming process of configuring and maintaining static routes.
- Disadvantages of dynamic routing include:
  - Part of a router's resources are dedicated for protocol operation, including CPU time and network link bandwidth.
  - Times when static routing is more appropriate.

Using Static Routing

Networks typically use a combination of both static and dynamic routing.

- Static routing has several primary uses:
- Providing ease of routing table maintenance in smaller networks that are not expected to grow significantly.
- Routing to and from a stub network. A network with only one default route out and no knowledge of any remote networks.
- Accessing a single default router. This is used to represent a path to any network that does not have a match in the routing table.

OSPF – Open Shortest Path First

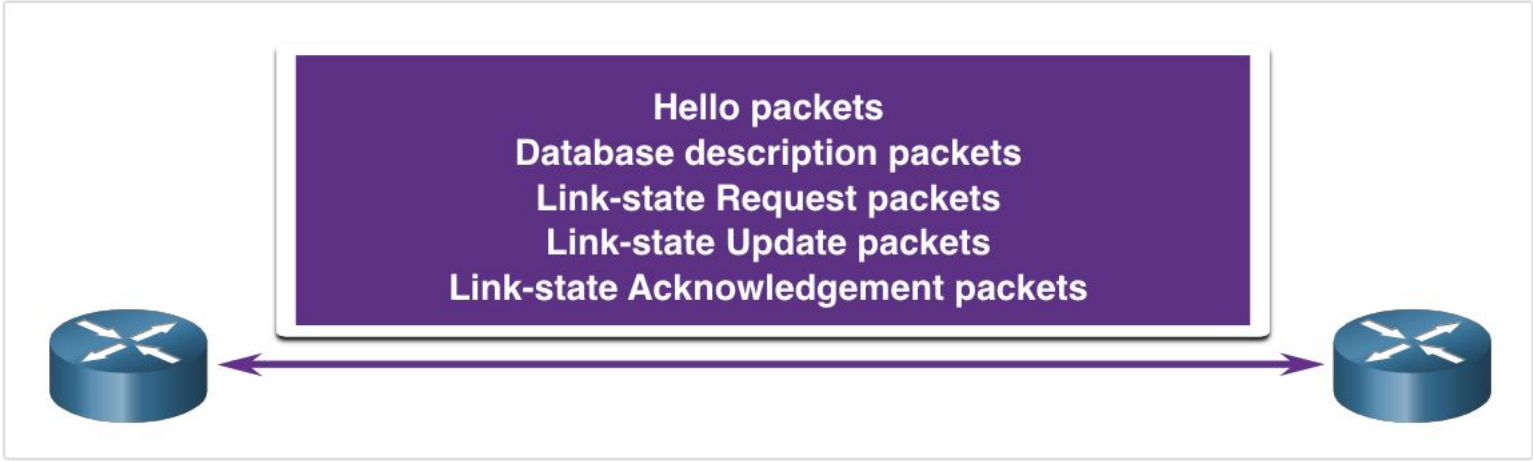
- Open Shortest Path First (OSPF), which includes single-area and multiarea. OSPFv2 is used for IPv4 networks.
- OSPFv3 is used for IPv6 networks.
- OSPF is a link-state routing protocol that uses the concept of areas.
- A network administrator can divide the routing domain into distinct areas that help control routing update traffic.
- A link is an interface on a router.
- A link is also a network segment that connects two routers, or a stub network such as an Ethernet LAN that is connected to a single router.
- Information about the state of a link is known as a link-state.
- All link-state information includes the network prefix, prefix length, and cost.

OSPF Components/ Routing Protocol Messages

Routers running OSPF exchange messages to convey routing information using five types of packets:

- Hello packet
- Database description packet
- Link-state request packet
- Link-state update packet
- Link-state acknowledgment packet

These packets are used to **discover** neighboring routers and also to **exchange routing information** to maintain accurate information about the network.



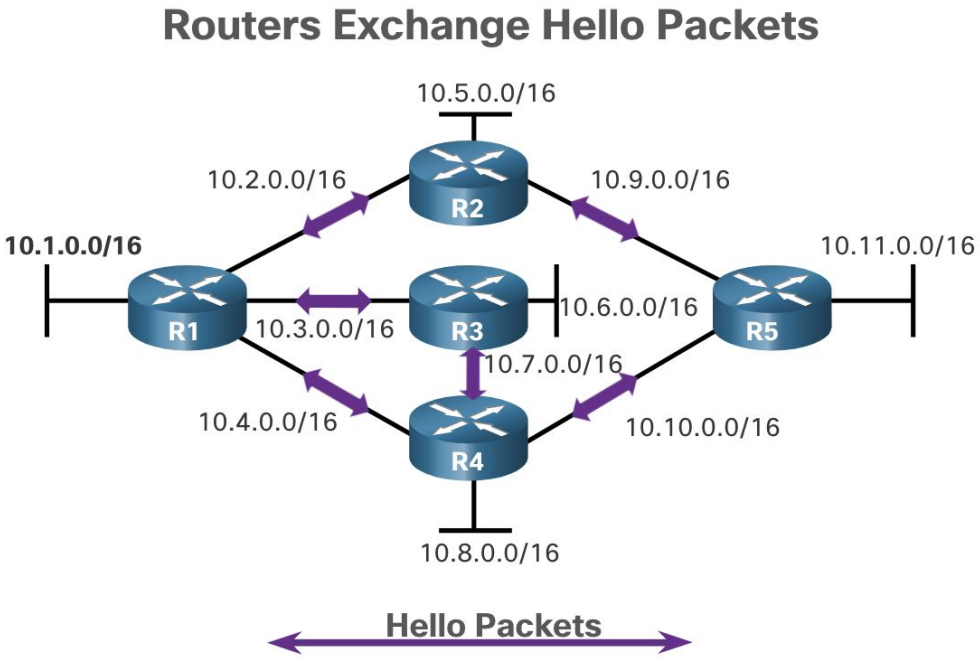
Link-State Operation

- *OSPF is a link-state routing protocol*
- To maintain routing information, OSPF routers complete a generic **link-state routing process** to reach a state of convergence.
- Each link between routers is labeled with a **cost value**.
- In OSPF, **cost** is used to determine **the best path** to the destination.
- The following are the link-state routing steps that are completed by a router:
  1. Establish Neighbor Adjacencies
  2. Exchange Link-State Advertisements
  3. Build the Link State Database
  4. Execute the SPF Algorithm
  5. Choose the Best Route

Link-State Operation/ Establishing Neighbor Adjacencies

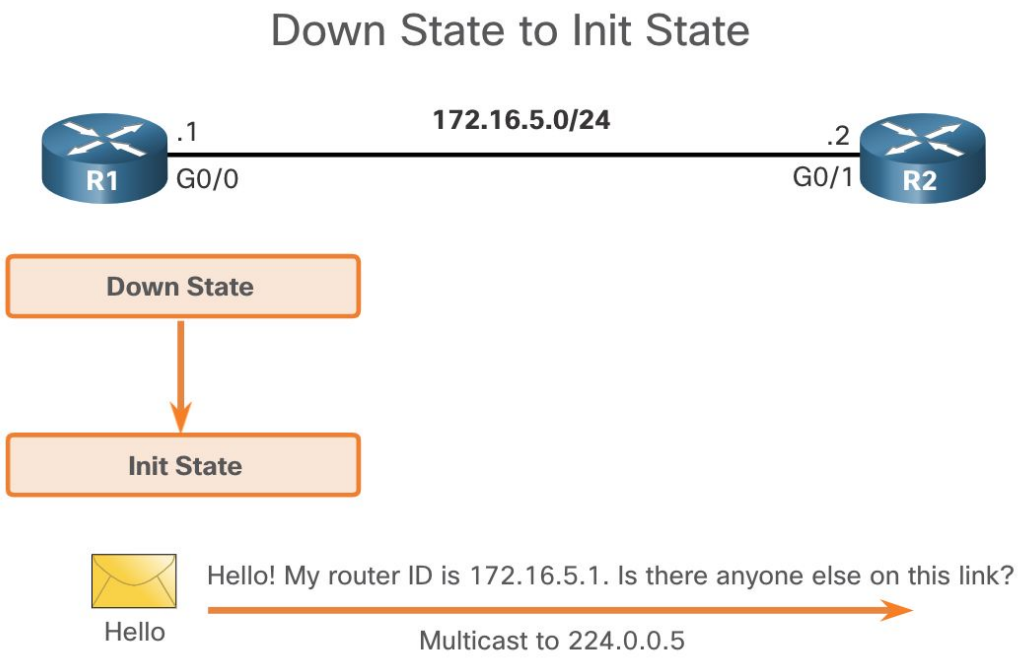
1. Establish Neighbor Adjacencies

- OSPF-enabled routers must recognize each other on the network before they can share information.
- An OSPF-enabled router sends **Hello packets** out all OSPF-enabled interfaces to determine if neighbors are present on those links.
- If a neighbor is present, the OSPF-enabled router attempts to establish a neighbor adjacency with that neighbor.



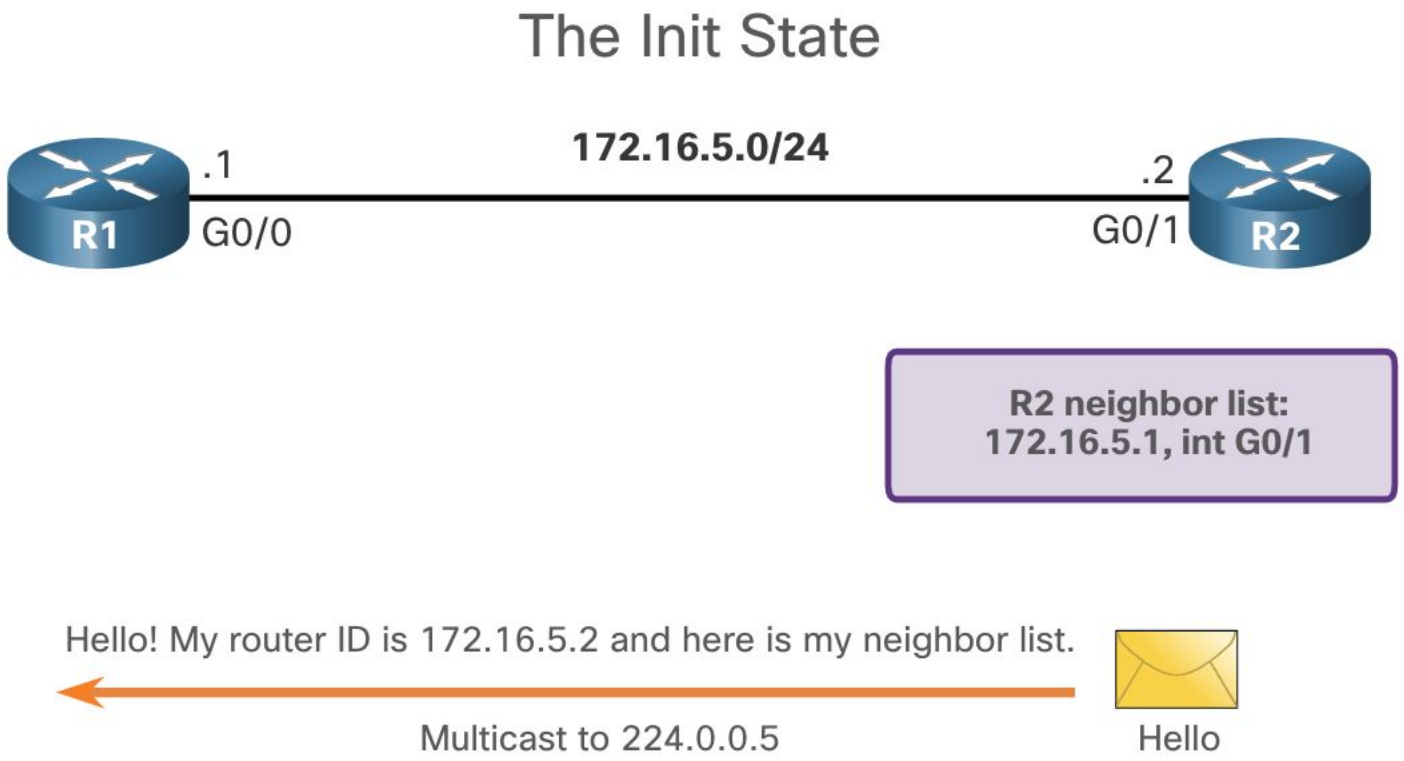
Establishing Neighbor Adjacencies

- When OSPF is enabled on an interface, the router must determine if there is another OSPF neighbor on the link by sending a **Hello packet** that contains its **router ID** out all OSPF-enabled interfaces.
- The Hello packet is sent to the reserved all OSPF Routers IPv4 multicast address 224.0.0.5.
- The OSPF router ID is a a 32-bit number that uniquely identifies each router in the OSPF area.
- When a neighboring OSPF-enabled router receives a **Hello packet** with a **router ID** that is not within its neighbor list, the receiving router attempts to establish an adjacency with the initiating router.



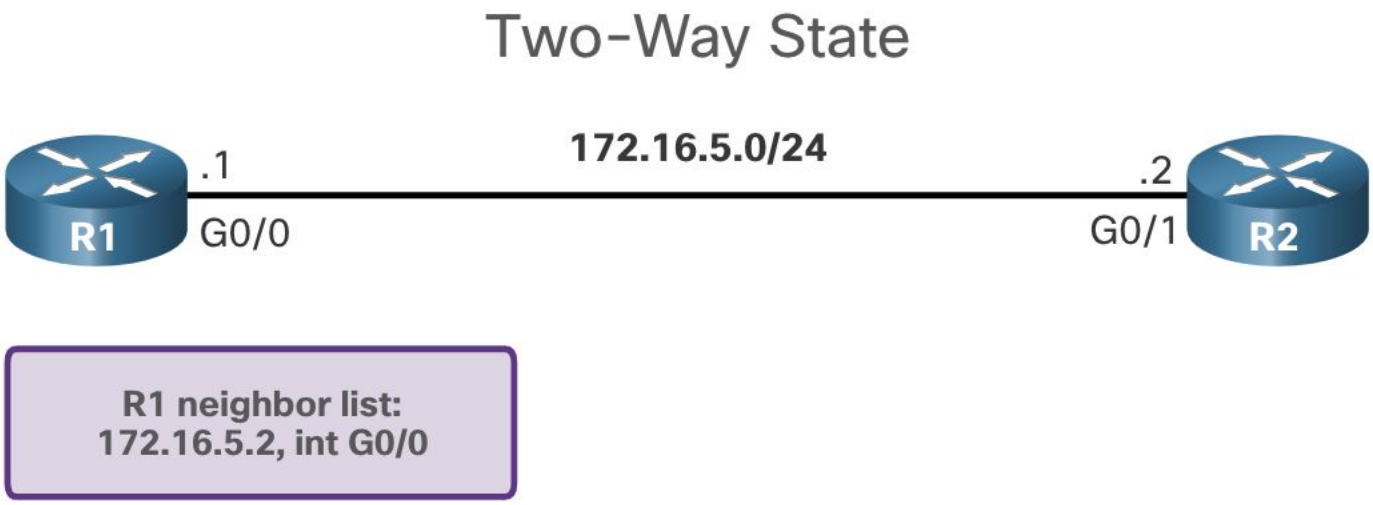
Establishing Neighbor Adjacencies (cont.)

- R2 receives the Hello packet from R1 and adds the R1 router ID to its neighbor list.
- R2 then sends a Hello packet to R1.
- The packet contains the R2 Router ID and the R1 Router ID in its list of neighbors on the same interface.



Establishing Neighbor Adjacencies (cont.)

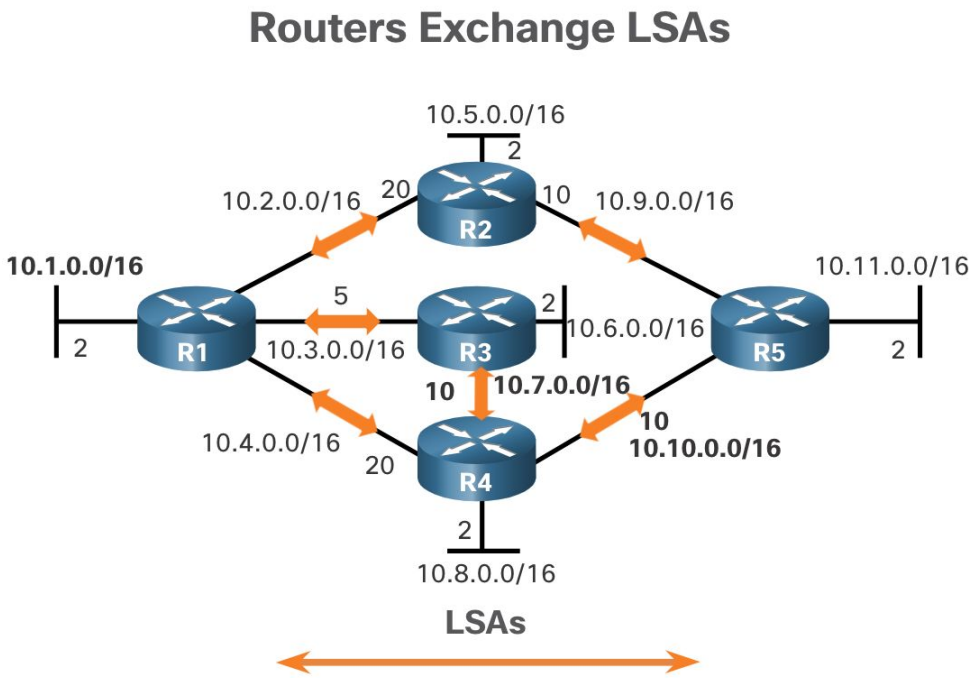
- R1 receives the Hello and adds the R2 Router ID to its list of OSPF neighbors.
- It also notices its own Router ID in the list of neighbors of the Hello packet.
- When a router receives a Hello packet with its Router ID listed in the list of neighbors, the router transitions from the Init state to the Two-Way state.



Link-State Operation/ Exchange Link-State Advertisements

2. Exchange Link-State Advertisements

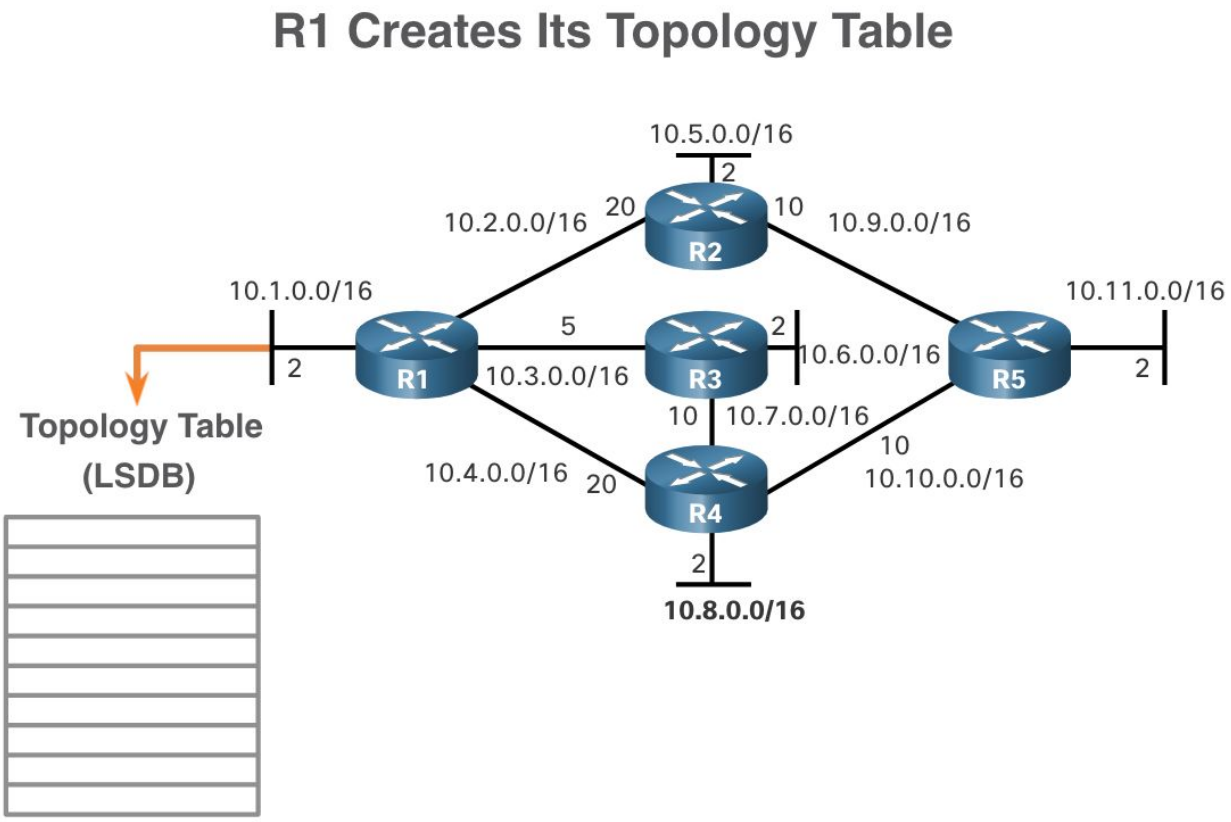
- After adjacencies are established, routers then exchange link-state advertisements (LSAs).
- LSAs contain the state and cost of each directly connected link.
- Routers flood their LSAs to adjacent neighbors.
- Adjacent neighbors receiving the LSA immediately flood the LSA to other directly connected neighbors, until all routers in the area have all LSAs.



Link-State Operation/ Build the Link State Database

3. Build the Link State Database

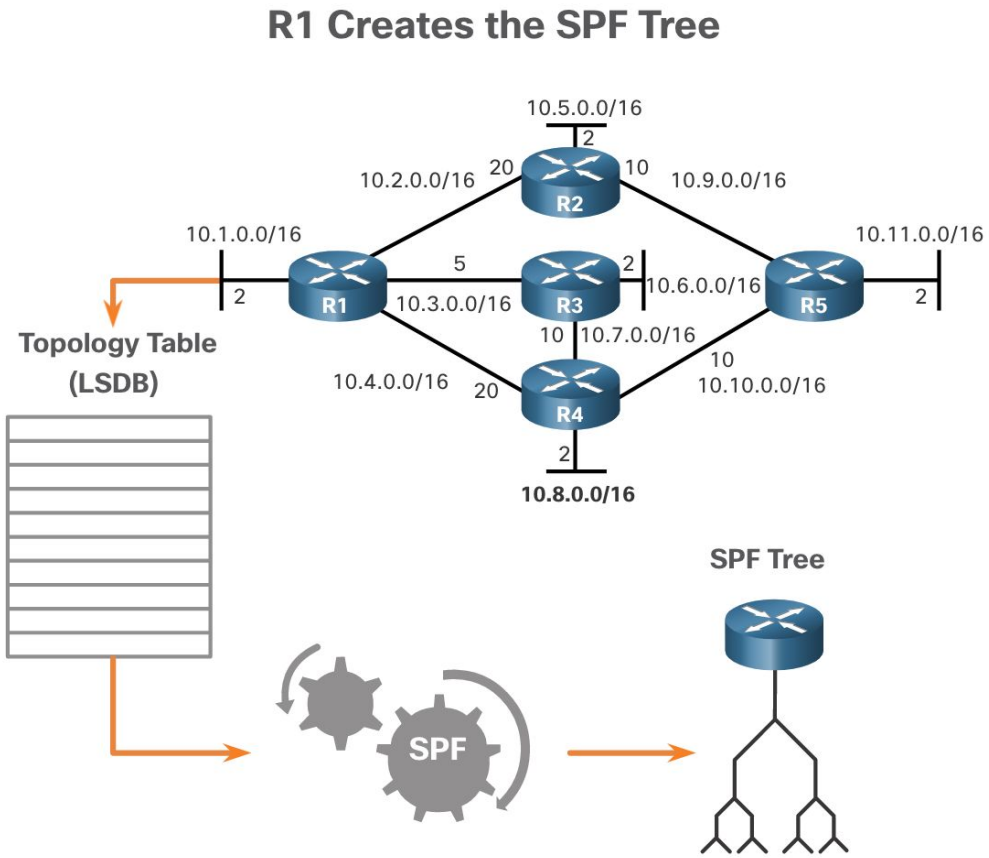
- After LSAs are received, OSPF-enabled routers build the topology table (LSDB).
- This database eventually holds all the information about the topology of the area.



Static Route/ Execute the SPF Algorithm

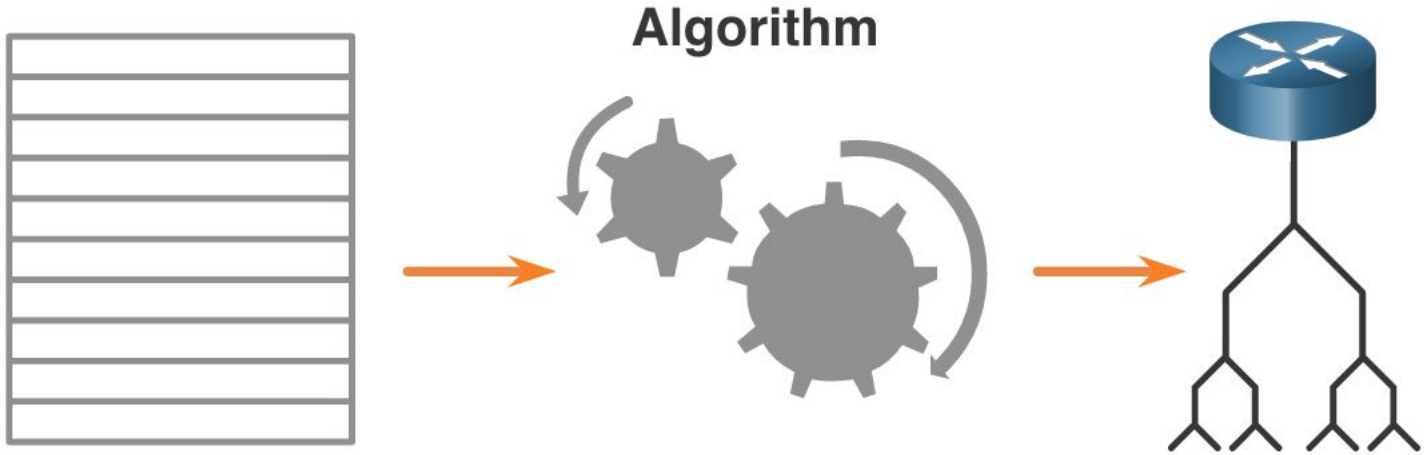
4. Execute the SPF Algorithm

- Routers then execute the SPF algorithm.
- The SPF algorithm creates the SPF tree.



OSPF Components/ Algorithm

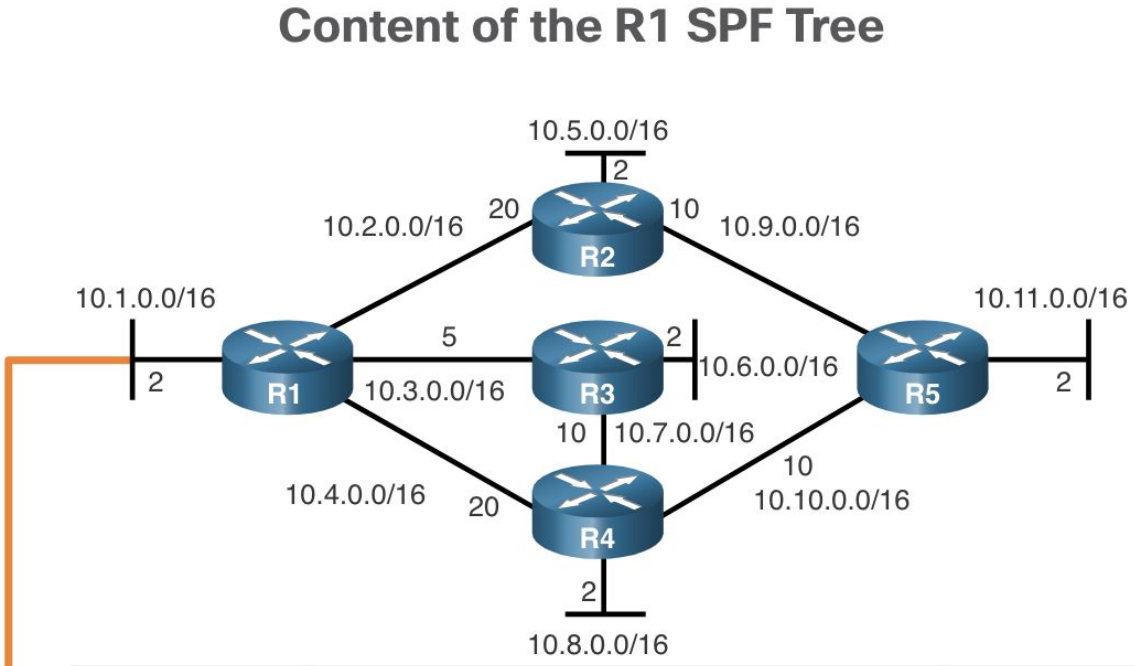
- The router builds the topology table using results of calculations based on the Dijkstra shortest-path first (SPF) algorithm.
- The SPF algorithm is based on the cumulative cost to reach a destination.
- The SPF algorithm creates an SPF tree by placing each router at the root of the tree and calculating the shortest path to each node.
- The SPF tree is then used to calculate the best routes.
- OSPF places the best routes into the forwarding database, which is used to make the routing table.



Static Route/ Choose the Best Route

5. Choose the Best Route

- After the SPF tree is built, the best paths to each network are offered to the IP routing table.
- The route will be inserted into the routing table unless there is a route source to the same network with a lower administrative distance, such as a static route.
- Routing decisions are made based on the entries in the routing table.



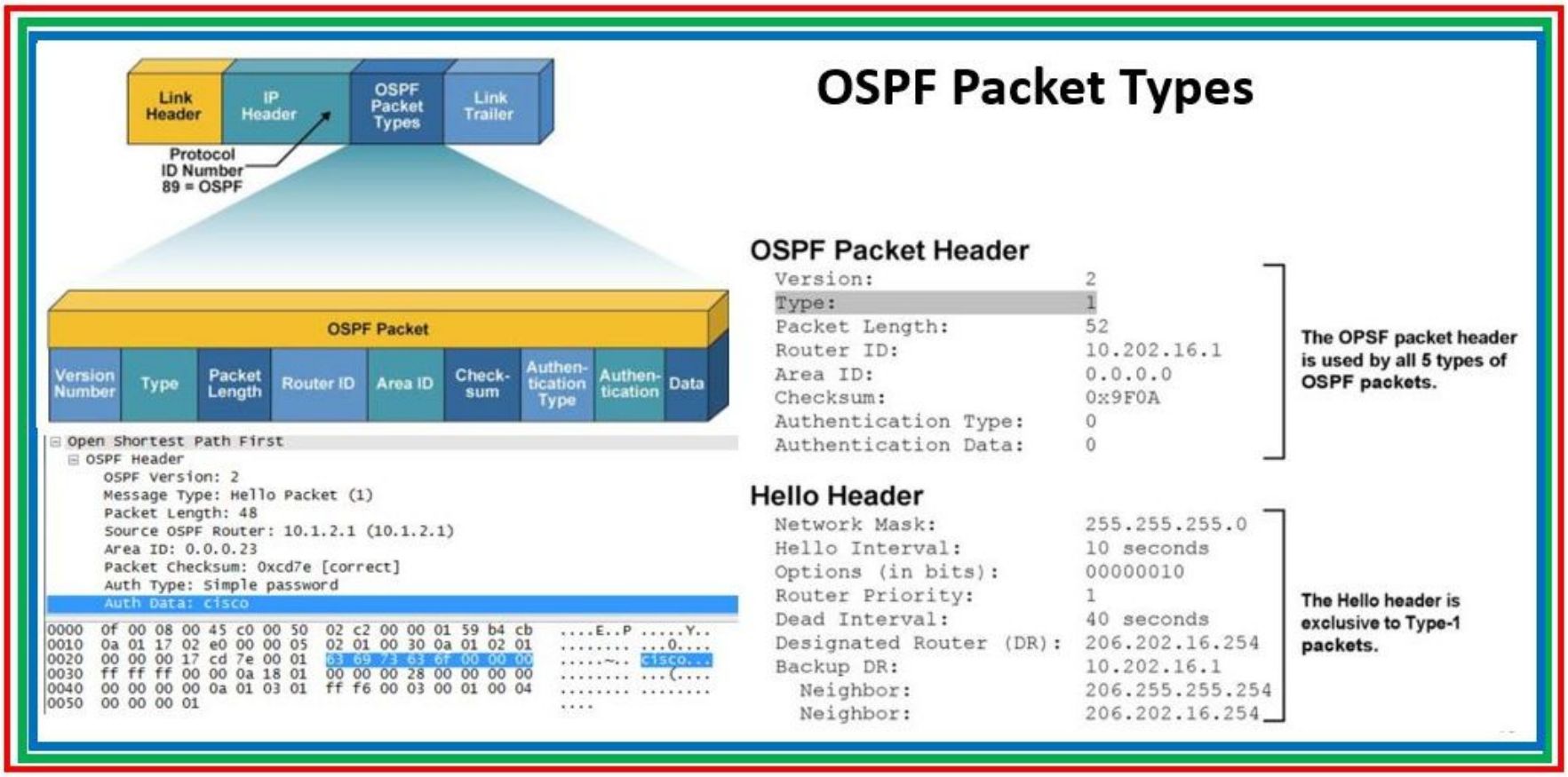
Destination	Shortest Path	Cost
10.5.0.0/16	R1→R2	22
10.6.0.0/16	R1→R3	7
10.7.0.0/16	R1→R3	15
10.8.0.0/16	R1→R3→R4	17
10.9.0.0/16	R1→R2	30
10.10.0.0/16	R1→R3→R4	25
10.11.0.0/16	R1→R3→R4→R5	27

Types of OSPF Packets

- Link-state packets are the tools used by OSPF to help determine the fastest available route for a packet.
- OSPF uses the following link-state packets (LSPs) to establish and maintain neighbor adjacencies and exchange routing updates.
- Each packet serves a specific purpose in the OSPF routing process, as follows:

Type	Packet Name	Description
1	Hello	Discovers neighbors and builds adjacencies between them
2	Database Description (DBD)	Checks for database synchronization between routers
3	Link-State Request (LSR)	Requests specific link-state records from router to router
4	Link-State Update (LSU)	Sends specifically requested link-state records
5	Link-State Acknowledgment (LSAck)	Acknowledges the other packet types

Hello and LSA packets



OSPF Components/ Data Structures

OSPF messages are used to create and maintain three OSPF databases, as follows:

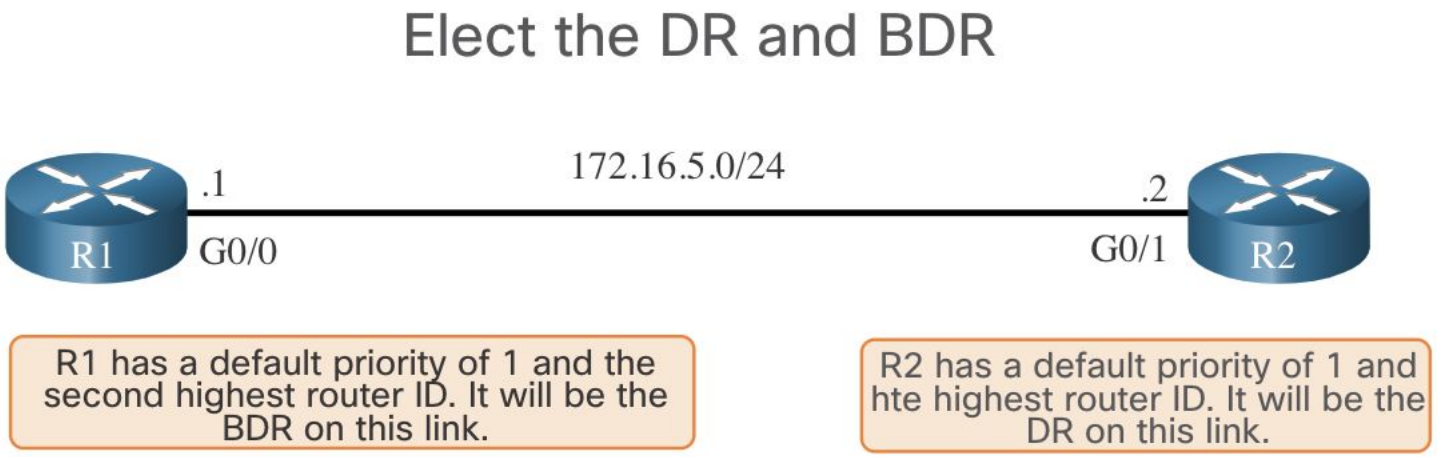
- Adjacency database - This creates the neighbor table.
- Link-state database (LSDB) - This creates the topology table.
- Forwarding database - This creates the routing table.

These tables maintained in RAM, contain a list of neighboring routers to exchange routing information.

Database	Table	Description
Adjacency Database	Neighbor Table	• List of all neighbor routers to which a router has established bidirectional communication. • This table is unique for each router. • Can be viewed using the **show ip ospf neighbor** command.
Link-state Database	Topology Table	• Lists information about all other routers in the network. • This database represents the network topology. • All routers within an area have identical LSDB. • Can be viewed using the **show ip ospf database** command.
Forwarding Database	Routing Table	• List of routes generated when an algorithm is run on the link-state database. • The routing table of each router is unique and contains information on how and where to send packets to other routers. • Can be viewed using the **show ip route** command.

Establishing Neighbor Adjacencies (cont.)

- Because R1 and R2 are interconnected over an Ethernet network, a DR and BDR election takes place.
- This process only occurs on multiaccess networks such as Ethernet LANs.
- Hello packets are continually exchanged to maintain router information.
- As shown in the figure, R2 becomes the DR and R1 is the BDR.



Single-Area and Multiarea OSPF

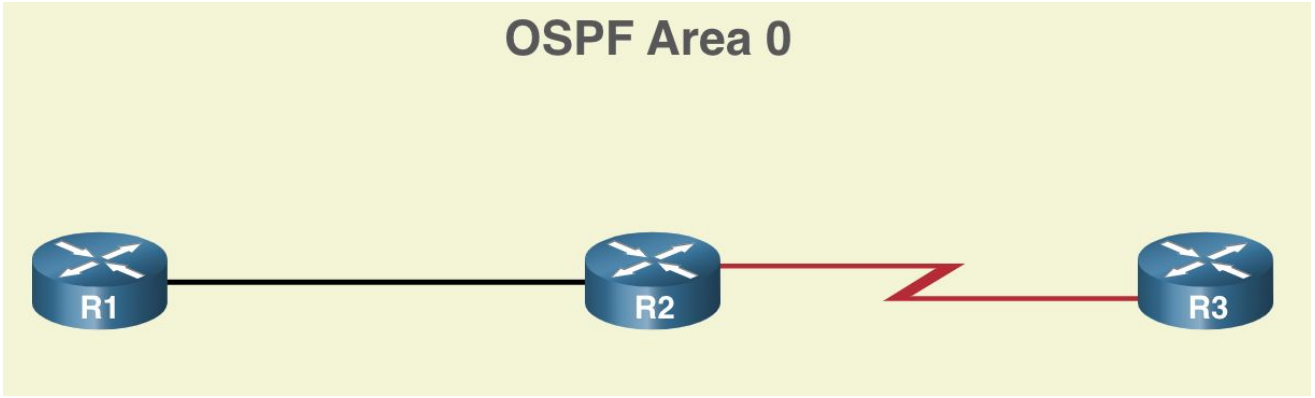
To make OSPF more efficient and scalable, OSPF supports hierarchical routing using areas.

An OSPF area is a group of routers that share the same link-state information in their LSDBs.

OSPF can be implemented in one of two ways, as follows:

- Single-Area OSPF - All routers are in one area. Best practice is to use area 0.
- Multiarea OSPF - OSPF is implemented using multiple areas, in a hierarchical fashion. All areas must connect to the backbone area (area 0). Routers interconnecting the areas are referred to as Area Border Routers (ABRs).

Single-Area OSPF

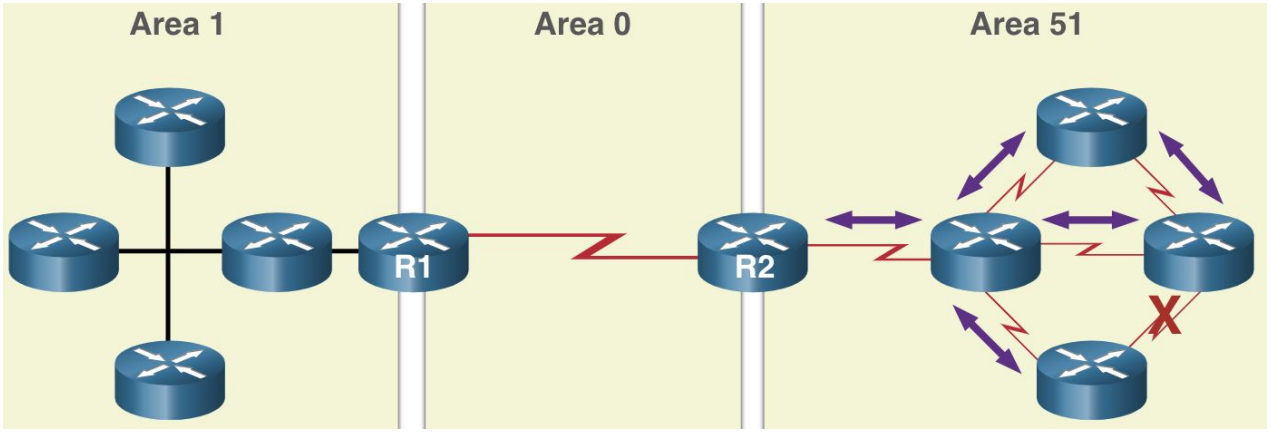


Multiarea OSPF

- One large routing domain can be divided into smaller areas, to support hierarchical routing.
- Routing still occurs between the areas (interarea routing).
- Routing operations, such as recalculating the database, are kept within an area.
- For instance, any time a router receives new information about a topology change within the area (including the addition, deletion, or modification of a link) the router must rerun the SPF algorithm, create a new SPF tree, and update the routing table.
- The SPF algorithm is CPU-intensive and the time it takes for calculation depends on the size of the area.
- Note: Routers in other areas receive updates regarding topology changes, but these routers only update the routing table, not rerun the SPF algorithm.

- Link failure affects the local area only (area 51).
- The ABR (R2) isolates the flooding of a specific LSA to area 51.
- Routers in areas 0 and 1 do not need to run the SPF algorithm.

Multi-Area OSPF



Purpose of Multiarea OSPF

Reduced routing overhead: By dividing the network into multiple areas, OSPF can limit the extent of link-state advertisements (LSAs) and reduce the amount of routing information exchanged between routers. *Each area maintains its own link-state database, and routers within an area only need to process LSAs relevant to that area*. This results in less routing traffic and lower resource consumption on routers.

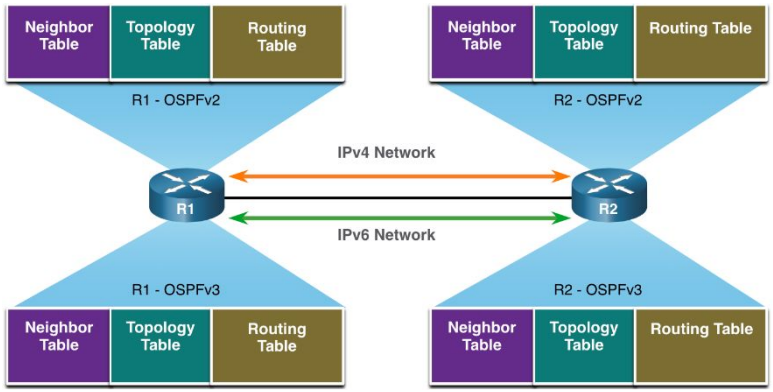
Faster convergence: In OSPF, when a topology change occurs, routers must recalculate their routing tables using the Shortest Path First (SPF) algorithm. In a large, single-area network, this process can be time-consuming and resource-intensive. Multi-Area OSPF confines the impact of topology changes within an area, allowing for faster convergence and reducing the load on routers.

Improved network stability: By confining the effects of network failures and changes within an area, Multi-Area OSPF enhances overall network stability. Additionally, the use of summarization at area borders can help prevent the propagation of suboptimal or unstable routes between areas.

Simplified network administration: Multi-Area OSPF enables network administrators to logically organize their networks, making it easier to manage, troubleshoot, and maintain. The hierarchical design also allows for the implementation of different routing policies and configurations within each area, providing increased flexibility and control.
Optimized routing: The hierarchical design of Multi-Area OSPF allows for route summarization at Area Border Routers (ABRs), which can reduce the number of routes in a router's routing table. This results in more efficient routing, lower memory consumption, and faster lookup times.

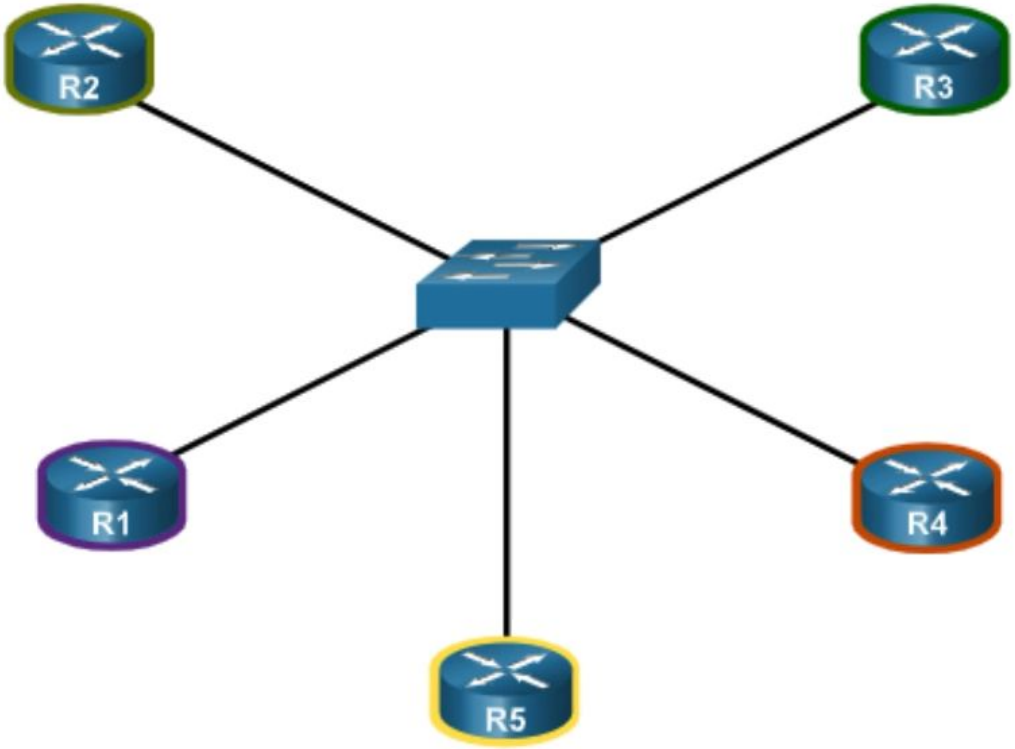
OSPFv3

- OSPFv3 is the OSPFv2 equivalent for exchanging IPv6 prefixes.
- Similar to its IPv4 counterpart, OSPFv3 exchanges routing information to populate the IPv6 routing table with remote prefixes.
- OSPFv3 Address Families feature, OSPFv3 includes support for both IPv4 and IPv6.
- OSPFv3 also uses the SPF algorithm as the computation engine to determine the best paths throughout the routing domain.
- OSPFv3 has separate processes from its IPv4 counterpart.
- The processes and operations are the same as in the IPv4 routing protocol, but run independently.
- OSPFv2 and OSPFv3 each have separate adjacency tables, OSPF topology tables, and IP routing tables.
- The OSPFv3 configuration and verification commands are similar to those used in OSPFv2.



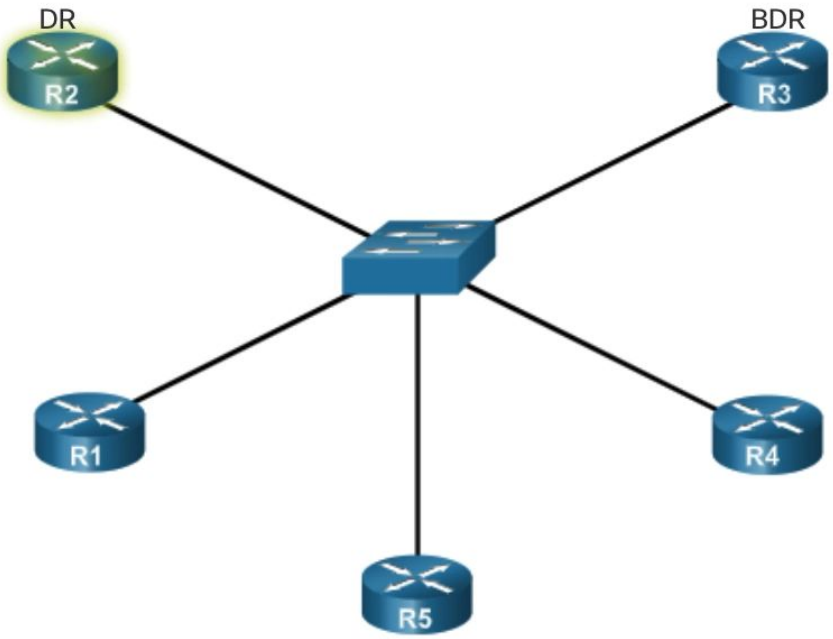
LSA Flooding Without a DR

- A dramatic increase in the number of routers also dramatically increases the number of LSAs exchanged between the routers.
- This flooding of LSAs significantly impacts the operation of OSPF.



LSA Flooding With a DR

- The solution to managing the number of adjacencies and the flooding of LSAs on a multiaccess network is the DR.
- On multiaccess networks, OSPF elects a DR to be the collection and distribution point for LSAs sent and received.
- A BDR is also elected in case the DR fails.



Router Configuration Mode for OSPF

- OSPFv2 is enabled using the {**router ospf** *process-id*} global configuration mode command.
- The *process-id* value represents a number between 1 and 65,535 and is selected by the network administrator.
- The *process-id* value is locally significant, which means that it does not have to be the same value on the other OSPF routers to establish adjacencies with those neighbors.
- It is considered best practice to use the same *process-id* on all OSPF routers.

```
R1(config)# router ospf 10
R1(config-router)# ?
  area                    OSPF area parameters
  auto-cost               Calculate OSPF interface cost according to bandwidth
  default-information     Control distribution of default information
  distance                Define an administrative distance
  exit                    Exit from routing protocol configuration mode
  log-adjacency-changes   Log changes in adjacency state
  neighbor                Specify a neighbor router
  network                 Enable routing on an IP network
  no                      Negate a command or set its defaults
  passive-interface       Suppress routing updates on an interface
  redistribute            Redistribute information from another routing protocol
  router-id               router-id for this OSPF process
R1(config-router)#
```

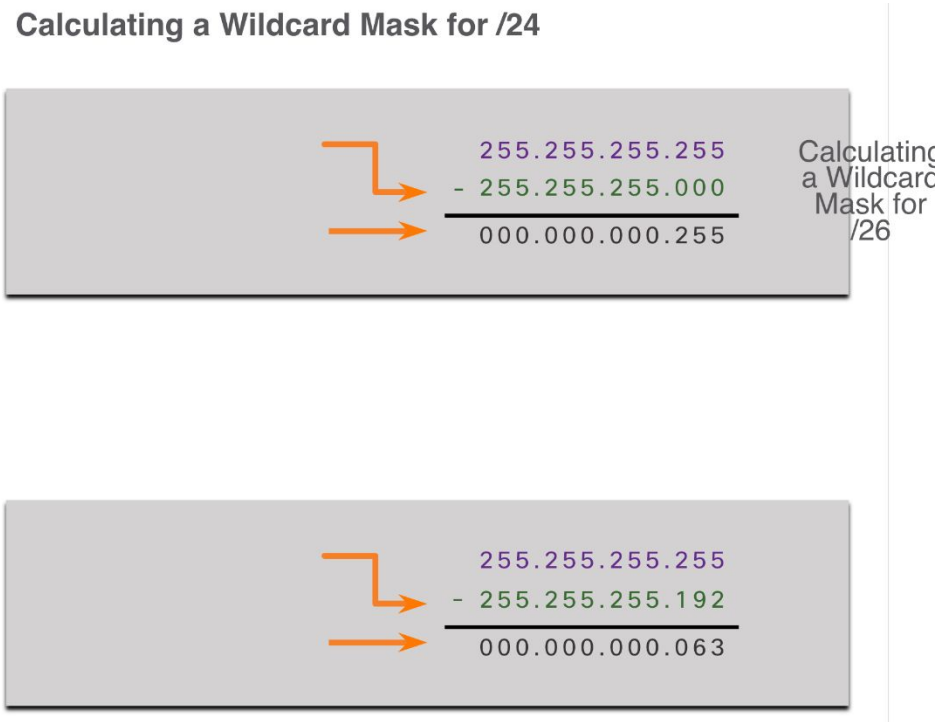
Router Configuration Mode for OSPF

- The *network-address wildcard-mask* syntax is used to enable OSPF on interfaces. Any interfaces on a router that match the network address in the **network** command are enabled to send and receive OSPF packets.
- The **area** *area-id* syntax refers to the OSPF area.
- When configuring single-area OSPFv2, the **network** command **must be** configured with the same *area-id* value on all routers.
- It is good practice to use an area ID of 0 with single-area OSPFv2.

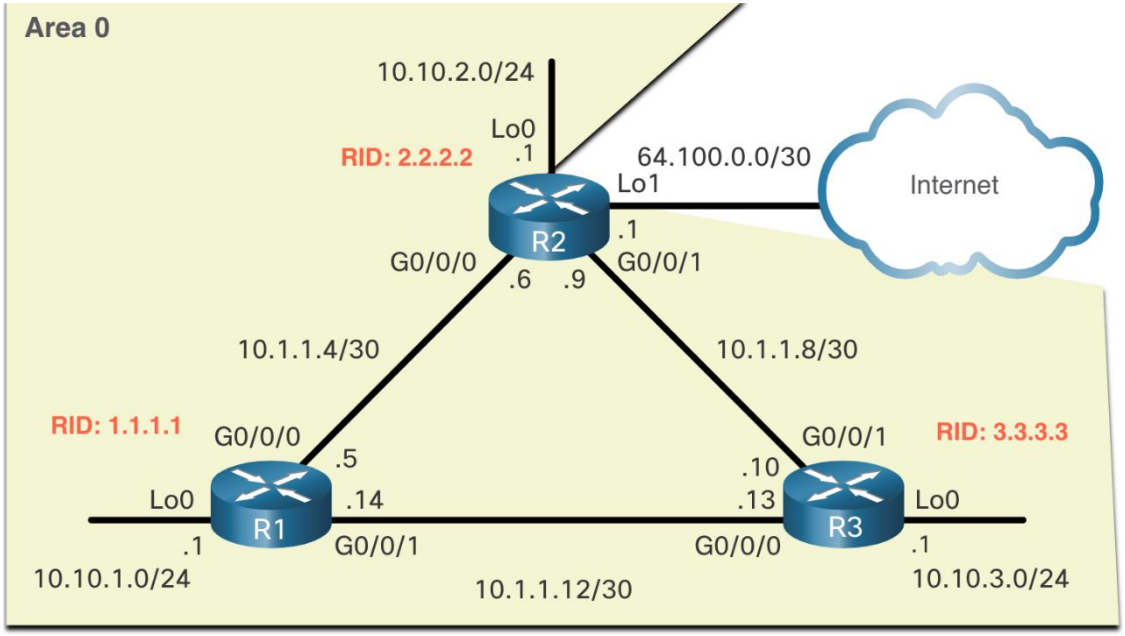
```
Router(config-router)# network network-address wildcard-mask area area-id
```

The Wildcard Mask

- The wildcard mask is typically the inverse of the subnet mask configured on that interface.
- In a subnet mask, binary 1 is equal to a match and binary 0 is not a match.
- In a wildcard mask, the reverse is true.
  - Wildcard mask bit 0 - Matches the corresponding bit value in the address.
  - Wildcard mask bit 1 - Ignores the corresponding bit value in the address.
- The easiest method for calculating a wildcard mask is to subtract the network subnet mask from 255.255.255.255, as shown for /24 and /26 subnet masks in the figure.



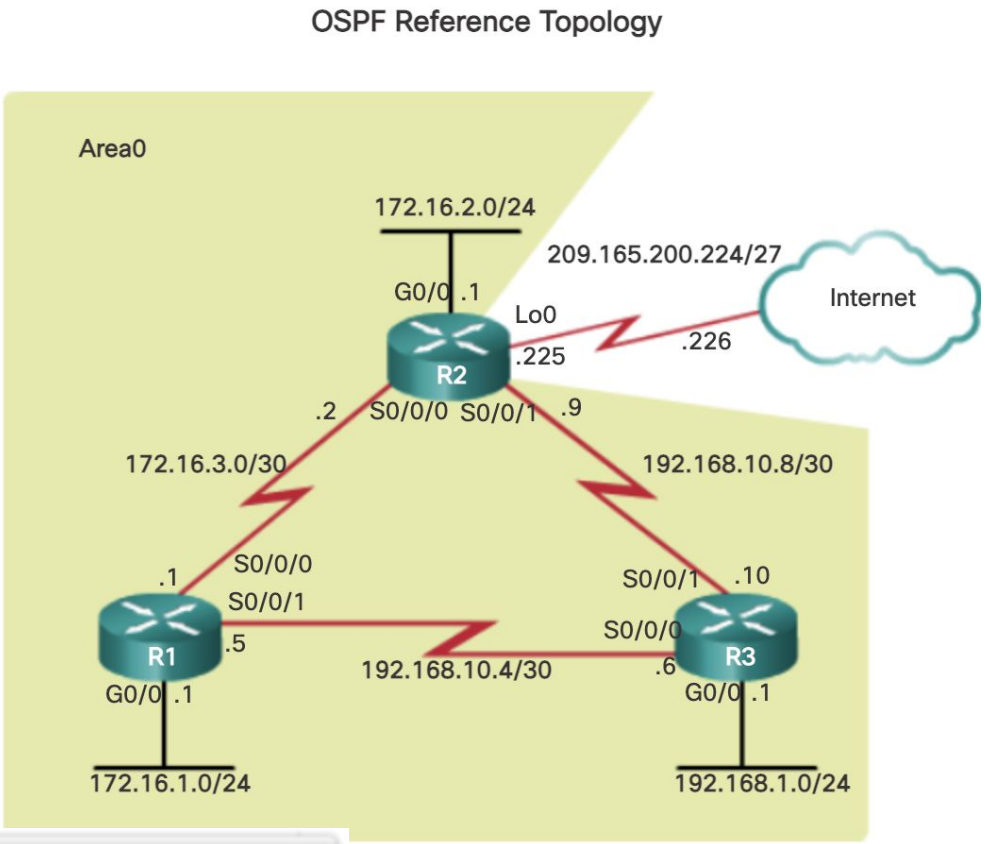
Configure a Router ID



- Use the **router-id** *rid* router configuration mode command to manually assign a router ID.
- In the example, the router ID 1.1.1.1 is assigned to R1.
- Use the **show ip protocols** command to verify the router ID.

```
R1(config)# router ospf 10
R1(config-router)# router-id 1.1.1.1
R1(config-router)# end
*May 23 19:33:42.689: %SYS-5-CONFIG_I: Configured from console by console
R1# show ip protocols | include Router ID
Router ID 1.1.1.1
R1#
```

OSPF Configuration




```
R1(config)# router ospf 10
R1(config-router)# network 172.16.1.0 0.0.0.255 area 0
R1(config-router)# network 172.16.3.0 0.0.0.3 area 0
R1(config-router)# network 192.168.10.4 0.0.0.3 area 0
R1(config-router)#
```

```
R1(config)# router ospf 10
R1(config-router)# network 172.16.1.1 0.0.0.0 area 0
R1(config-router)# network 172.16.3.1 0.0.0.0 area 0
R1(config-router)# network 192.168.10.5 0.0.0.0 area 0
R1(config-router)#
```

Passive Interface

- By default, OSPF messages are forwarded out all OSPF-enabled interfaces.
- These messages really only need to be sent out interfaces that are connecting to other OSPF-enabled routers.
- Sending out unneeded messages on a LAN affects the network in three ways, as follows:
  - Inefficient Use of Bandwidth
  - Inefficient Use of Resources
  - Increased Security Risk

```
R1(config)# router ospf 10
R1(config-router)# passive-interface GigabitEthernet 0/0
R1(config-router)# end
R1#
```

Teaser/ Monty Hall Problem

COURSE: **NETWORK ADMINISTRATION AND MANAGEMENT**

COURSE INSTRUCTOR: **MIRALDA CUKA, PHD**

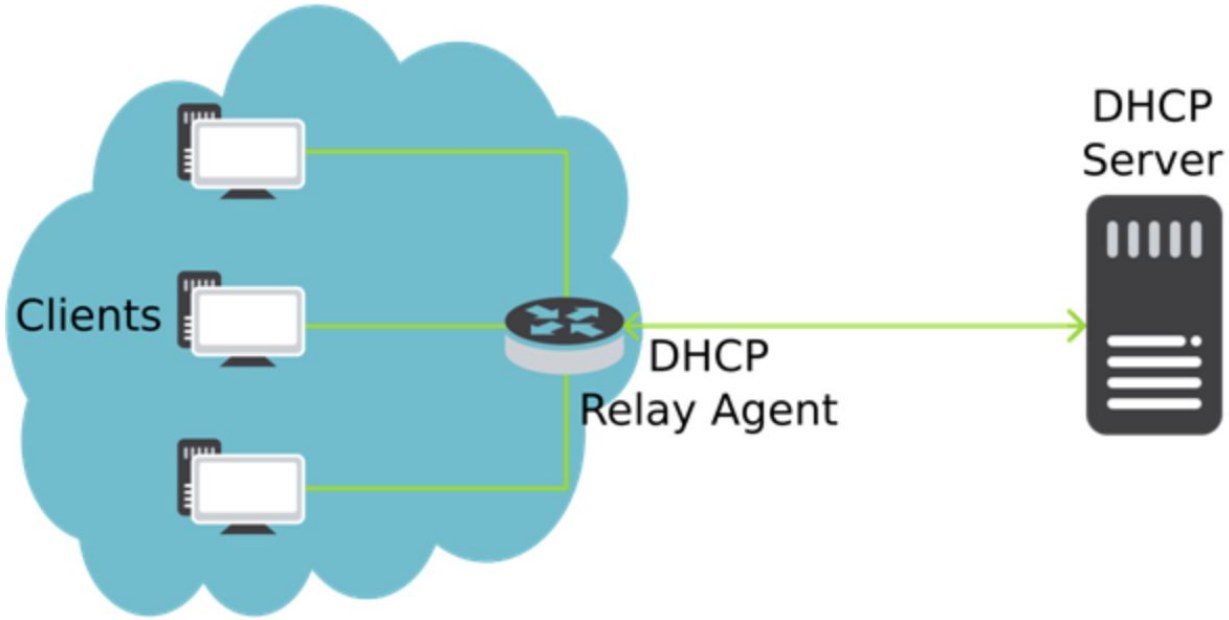
Lecture 7
DHCP Configuration

IP Allocation

- Every device that connects to a network needs a unique IP address.
- Network administrators can assign static IP addresses to routers, servers, printers, and other network devices.
- Static addresses enable administrators to manage these devices remotely.
- It is easier for network administrators to access a device when they can easily determine its IP address.
- However, computers and users in an organization often change locations, physically and logically.
- It is difficult and time consuming to assign new IP addresses every time a host moves.

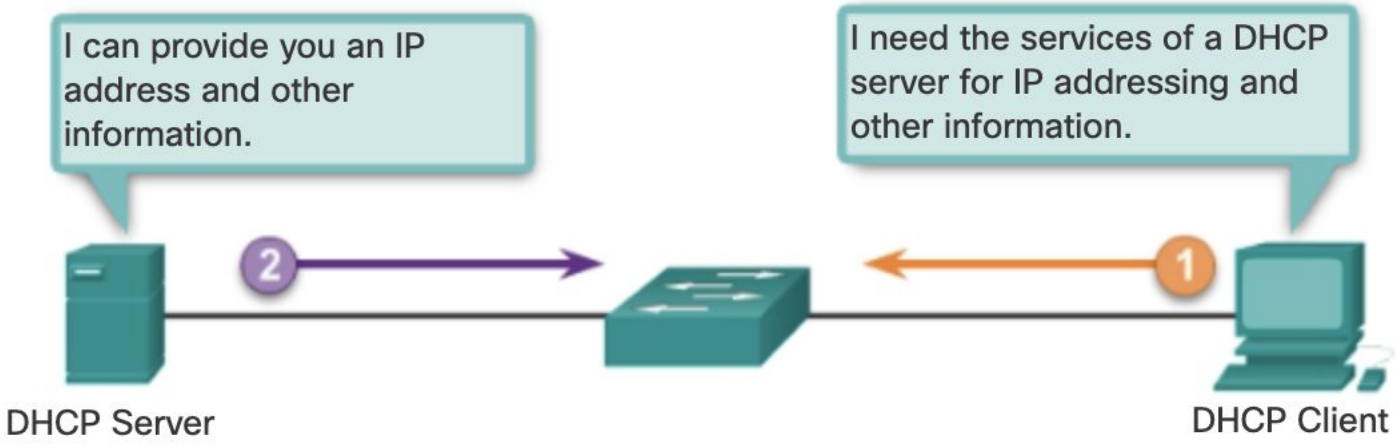
Dynamic Host Configuration Protocol (DHCP)

- A Dynamic Host Configuration Protocol (DHCP) server to the local network simplifies IP address assignment to both desktop and mobile devices.
- Using a centralized DHCP server enables organizations to administer all dynamic IP address assignments from a single server.
- This practice makes IP address management more effective.
- DHCP is available for both IPv4 (DHCPv4) and for IPv6 (DHCPv6).



Dynamic Host Configuration Protocol (DHCP) (cont.)

- DHCP servers assigns IP addresses and other network configuration information dynamically.
- DHCP is an extremely useful and timesaving tool for network administrators.
- A dedicated DHCP server is scalable and relatively easy to manage.
- However, for a small network, a router can be configured to provide DHCP services without the need for a dedicated server.



DHCPv4 Address Allocation Mechanisms

- Manual Allocation - The administrator assigns a pre-allocated IP address to the client, and DHCP communicates only the IP address to the device.
- Automatic Allocation - DHCP automatically assigns a static IP address permanently to a device, selecting it from a pool of available addresses. There is no lease and the address is permanently assigned to the device.
- Dynamic Allocation - DHCPv4 dynamically assigns, or leases, an IPv4 address from a pool of addresses for a limited period of time chosen by the server, or until the client no longer needs the address.

DHCP Operation

- DHCP works in a client/server mode.
- When a client communicates with a DHCP server, the server assigns or leases an IP address to that client.
- The client connects to the network with that leased IP address until the lease expires.
- The client must contact the DHCP server periodically to extend the lease.
- This lease mechanism ensures that clients that move or power off do not keep addresses that they no longer need.
- When a lease expires, the DHCP server returns the address to the pool where it can be reallocated as necessary.

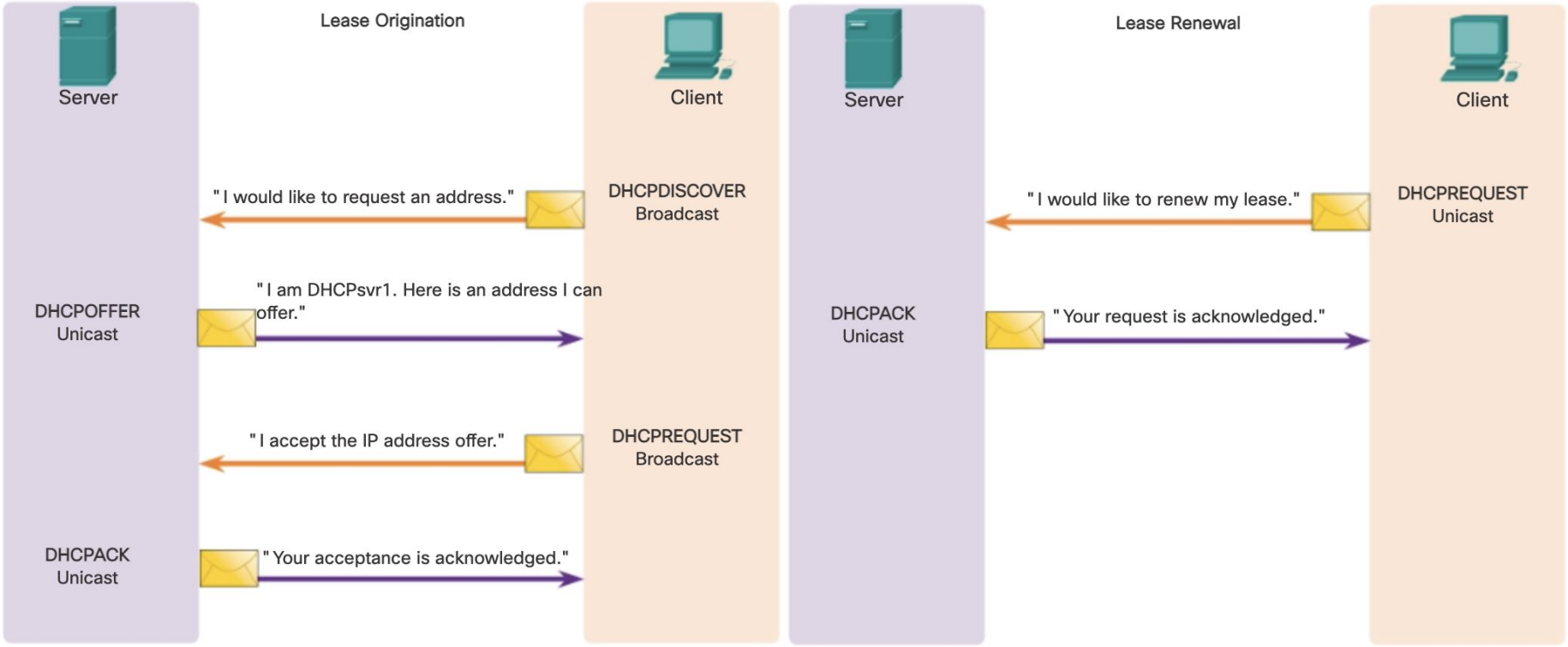
DHCP Operation

A client obtains a lease from a DHCP server.

1. DHCP Discover (DHCPDISCOVER)
2. DHCP Offer (DHCPOFFER)
3. DHCP Request (DHCPREQUEST)
4. DHCP Acknowledgment (DHCPACK)

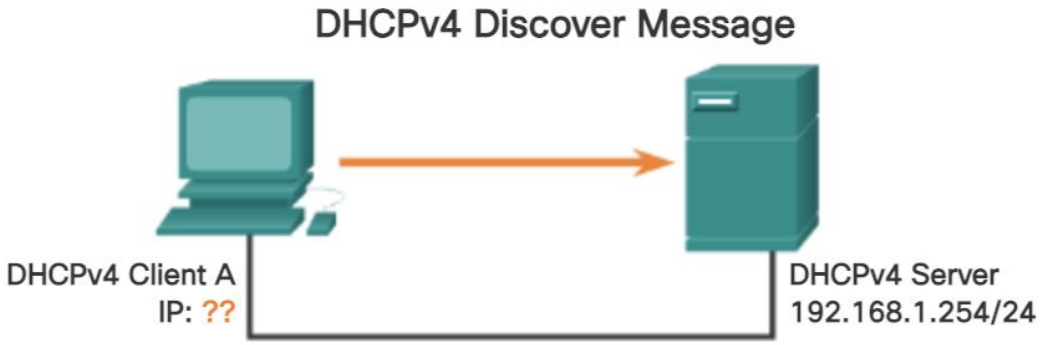
A client renews its lease from a DHCP server.

1. DHCP Request (DHCPREQUEST)
2. DHCP Acknowledgment (DHCPACK)



The client and server send acknowledgment messages, and the process is complete.

DHCP Discover Message

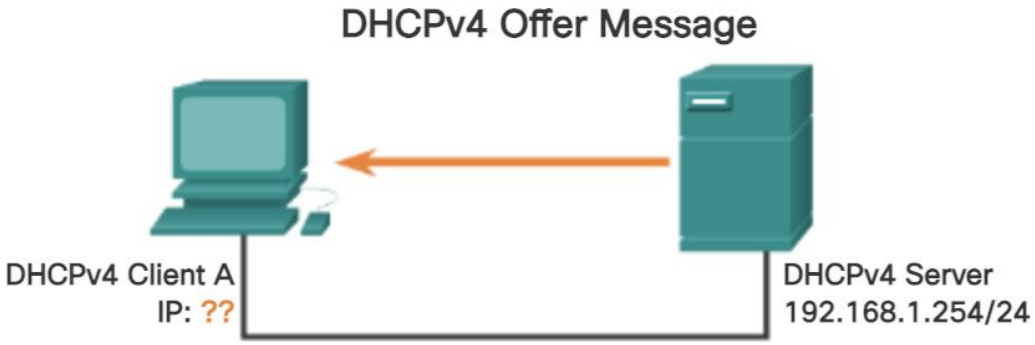


Ethernet Frame	IP	UDP	DHCPDISCOVER
DST MAC: FF:FF:FF:FF:FF:FF SRC MAC: MAC A	IP SRC: 0.0.0.0 IP DST: 255.255.255.255	UDP 67	CIADDR: 0.0.0.0 GIADDR: 0.0.0.0 Mask: 0.0.0.0 CHADDR: MAC A

MAC: Media Access Control Address
CIADDR: Client IP Address
GIADDR: Gateway IP Address
CHADDR: Client Hardware Address

The DHCP client sends a directed IP broadcast with a DHCPDISCOVER packet. In this example, the DHCP server is on the same segment and will pick up this request. The server notes the GIADDR field is blank; therefore, the client is on the same segment. The server also notes the hardware address of the client in the request packet.

DHCP Offer Message



Ethernet Frame	IP	UDP	DHCP Reply
DST MAC: MAC A SRC MAC: MAC Serv	IP SRC: 192.168.1.254 IP DST: 192.168.1.10	UDP 68	CIADDR: 192.168.1.10 GIADDR: 0.0.0.0 Mask: 255.255.255.0 CHADDR: MAC A

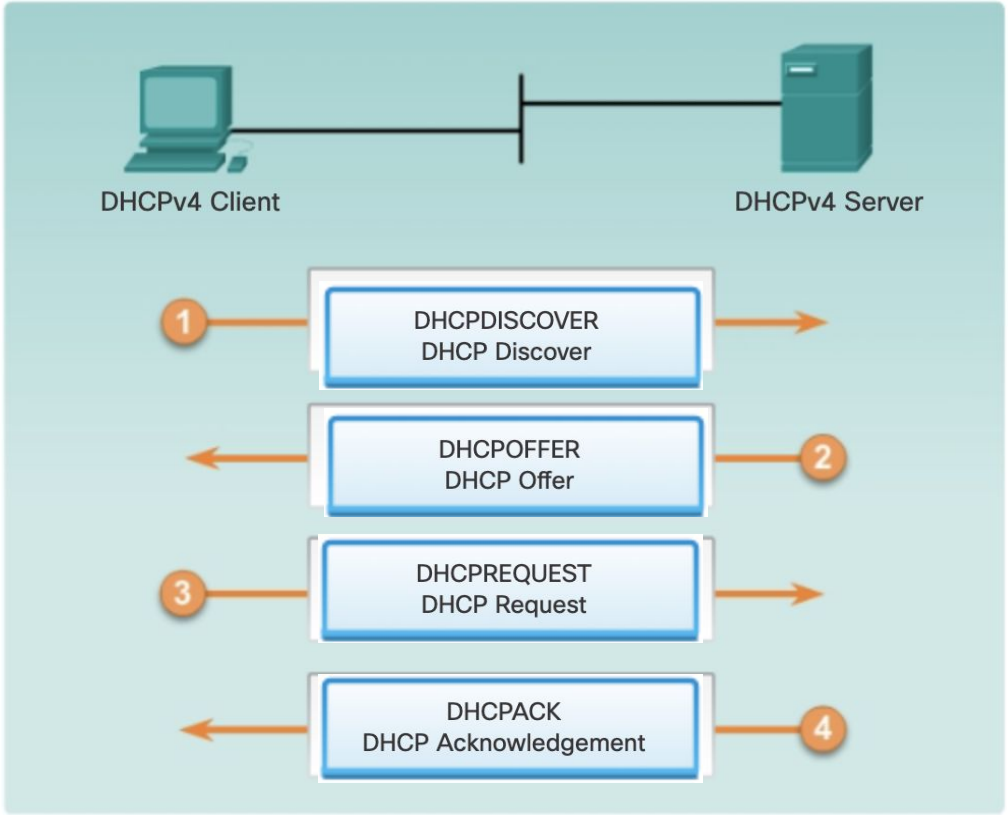
MAC: Media Access Control Address
CIADDR: Client IP Address
GIADDR: Gateway IP Address
CHADDR: Client Hardware Address

The DHCP server picks an IP address from the available pool for that segment, as well as the other segment and global parameters. The DHCP server puts them into the appropriate fields of the DHCP packet. The DHCP server then uses the hardware address of A (in CHADDR) to construct an appropriate frame to send back to the client.

DHCP Operation Example

Activity - Identify the Steps in DHCPv4 Operation

Order the steps to illustrate a DHCPv4 lease origination. Drag each of the DHCPv4 lease origination steps to its appropriate field, based on the circled numbers in the graphic.



DHCP Configuration Step 1: Excluding addresses

- The router functioning as the DHCPv4 server assigns all IPv4 addresses in a DHCPv4 address pool unless configured to exclude specific addresses.
- Some IPv4 addresses in a pool are assigned to network devices that require static address assignments.
- To exclude specific addresses, use the command:

  ip dhcp excluded address
- A single address or a range of addresses can be excluded by specifying the low-address and high-address of the range.
- Excluded addresses should include the addresses assigned to routers, servers, printers, and other devices that have been manually configured.

```
R1(config)# ip dhcp excluded-address low-address [high-address]
```

```
R1(config)# ip dhcp excluded-address 192.168.10.1 192.168.10.9
R1(config)# ip dhcp excluded-address 192.168.10.254
```

Range of addresses excluded

Single address excluded

DHCP Configuration Step 2: DHCP Pool

Configuring a DHCPv4 Pool

- Configuring a DHCPv4 server involves defining a pool of addresses to assign.
- The command:

ip dhcp pool *pool-name*

creates a pool with the specified name and puts the router in DHCPv4 configuration mode, which is identified by this prompt Router(dhcp-config)#.

```
R1(config)# ip dhcp pool pool-name
R1(dhcp-config)#
```

```
R1(config)# ip dhcp pool LAN-POOL-1
R1(dhcp-config)#
```

DHCP Configuration Step 3: Configuring Specific Tasks

Completing the DHCPv4 configuration

- The **address pool** and **default gateway router** must be configured.
- The **network** statement defines the range of available addresses.
- The **default-router** command defines the default gateway router.
- The gateway is the LAN interface of the router closest to the client devices. One gateway is required, but you can list up to eight addresses if there are multiple gateways.

Required Tasks	Command
Define the address pool.	**network** *network-number* [*mask* | */prefix-length*]
Define the default router or gateway.	**default-router** *address* [*address2...address8*]

Other Optional Commands

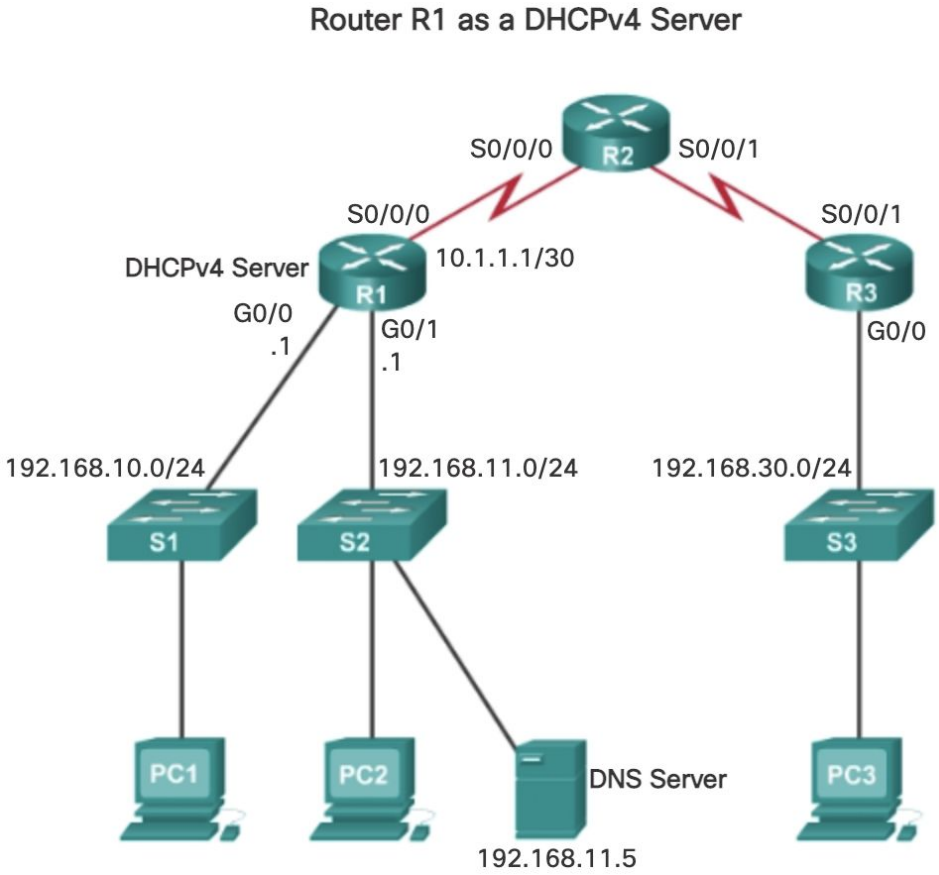
- Other DHCPv4 pool commands are optional.
- For example, the IPv4 address of the DNS server that is available to a DHCPv4 client is configured using the **dns-server** command.
- The **domain-name** *domain* command is used to define the domain name. The duration of the DHCPv4 lease can be changed using the **lease** command.

  The **netbios-name-server** command is used to define the NetBIOS WINS server.

Optional Tasks	Command
Define a DNS server.	**dns-server** *address [address2...address8]*
Define the domain name.	**domain-name** *domain*
Define the duration of the DHCP lease.	**lease** {*days [hours] [minutes] | infinite*}
Define the NetBIOS WINS server.	**netbios-name-server** *address [address2...address8]*

DHCP Configuration Example

- A sample configuration with basic DHCPv4 parameters configured on router R1, a DHCPv4 server for the 192.168.10.0/24 LAN.



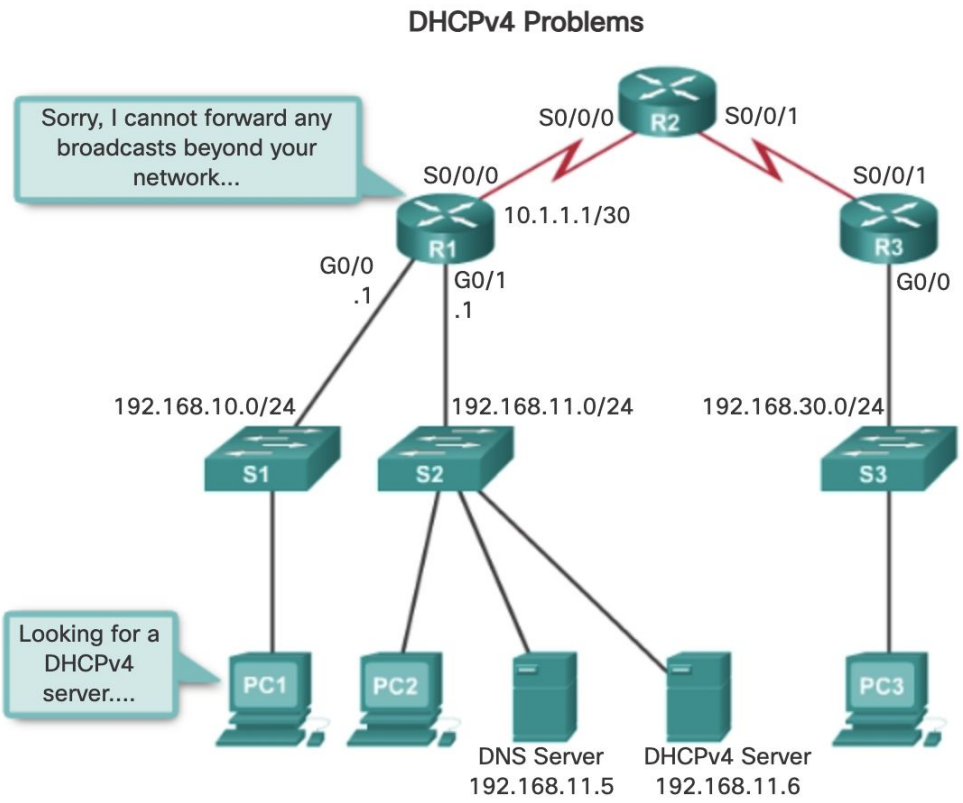

```
R1(config)# ip dhcp excluded-address 192.168.10.1 192.168.10.9
R1(config)# ip dhcp excluded-address 192.168.10.254
R1(config)# ip dhcp pool LAN-POOL-1
R1(dhcp-config)# network 192.168.10.0 255.255.255.0
R1(dhcp-config)# default-router 192.168.10.1
R1(dhcp-config)# dns-server 192.168.11.5
R1(dhcp-config)# domain-name example.com
R1(dhcp-config)# end
R1#
```

Disabling DHCPv4

- The DHCPv4 service is enabled, by default.
- To disable the service, **no service dhcp** command is used global configuration mode command.
- Using the **service dhcp** global configuration mode command to re-enable the DHCPv4 server process.
- Enabling the service has no effect if the parameters are not configured.

DHCP Relay

- In a complex hierarchical network, enterprise servers are usually located in a server farm.
- These servers may provide DHCP, DNS, TFTP, and FTP services for the network.
- Network clients are not typically on the same subnet as those servers.
- In order to locate the servers and receive services, clients often use broadcast messages.



DHCP Relay (cont.)

- A solution is to configure a *helper address*.
- This solution enables a router to act as a relay agent, forwards DHCPv4 broadcasts to the DHCPv4 server.
- When PC1 broadcasts a request to locate a DHCPv4 server, if R1 was configured as a DHCPv4 relay agent, it would forward the request to the DHCPv4 server located on subnet 192.168.11.0.
- The interface on R1 receiving the broadcast is configured with the **ip helper-address** interface configuration mode command.

```
R1(config)# interface g0/0
R1(config-if)# ip helper-address 192.168.11.6
R1(config-if)# end
R1# show ip interface g0/0
GigabitEthernet0/0 is up, line protocol is up
  Internet address is 192.168.10.1/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is 192.168.11.6
<output omitted>
```

Router as e DHCP Client

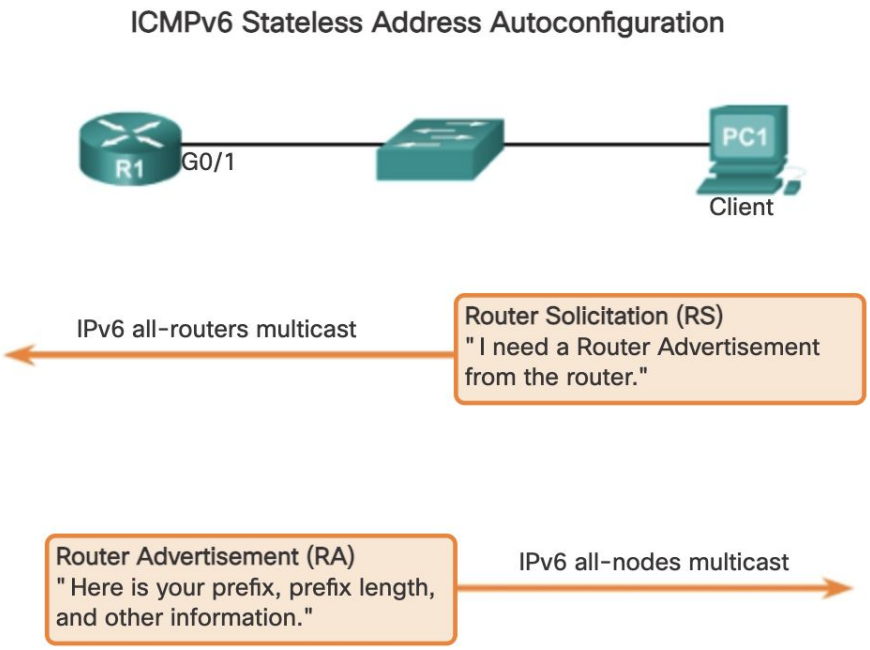
- Sometimes routers have to be configured as DHCPv4 clients in a similar manner to client computers.
- The method used depends on the ISP.
- To configure an Ethernet interface as a DHCP client, the **ip address dhcp** command on the interface is used.
- In the figure below, assume that an ISP has been configured to provide select customers with IP addresses from the 209.165.201.0/27 network range.
- After the G0/1 interface is configured with the **ip address dhcp** command, the **show ip interface g0/1** command confirms that the interface is up and that the address was allocated by a DHCPv4 server.


```
SOHO(config)# interface g0/1
SOHO(config-if)# ip address dhcp
SOHO(config-if)# no shutdown
SOHO(config-if)#
*Jan 31 17:31:11.507: %DHCP-6-ADDRESS_ASSIGN: Interface
GigabitEthernet0/1 assigned DHCP address 209.165.201.12, mask
255.255.255.224, hostname SOHO
SOHO(config-if)# end
SOHO# show ip interface g0/1
GigabitEthernet0/1 is up, line protocol is up
  Internet address is 209.165.201.12/27
  Broadcast address is 255.255.255.255
  Address determined by DHCP

<output omitted>
```

IPv6 Allocation

- Similar to IPv4, IPv6 global unicast addresses can be configured manually or dynamically.
- There are two methods in which IPv6 global unicast addresses can be assigned dynamically:
  - Stateless Address Autoconfiguration (SLAAC)
  - Dynamic Host Configuration Protocol for IPv6 (Stateful DHCPv6)



OSPF Components/ Data Structures

OSPF messages are used to create and maintain three OSPF databases, as follows:

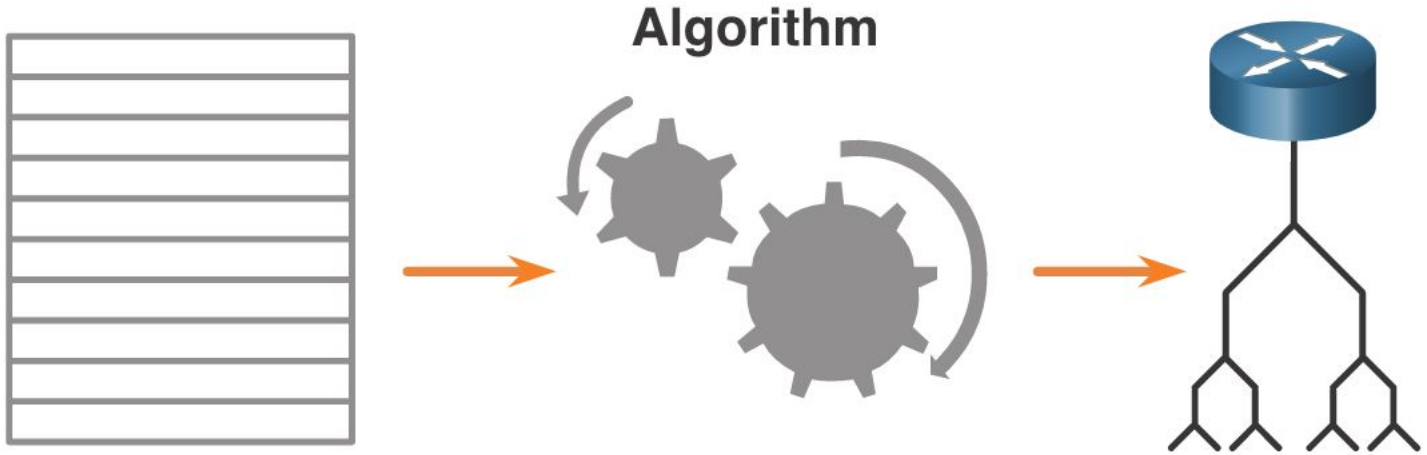
- Adjacency database - This creates the neighbor table.
- Link-state database (LSDB) - This creates the topology table.
- Forwarding database - This creates the routing table.

These tables maintained in RAM, contain a list of neighboring routers to exchange routing information.

Database	Table	Description
Adjacency Database	Neighbor Table	• List of all neighbor routers to which a router has established bidirectional communication. • This table is unique for each router. • Can be viewed using the **show ip ospf neighbor** command.
Link-state Database	Topology Table	• Lists information about all other routers in the network. • This database represents the network topology. • All routers within an area have identical LSDB. • Can be viewed using the **show ip ospf database** command.
Forwarding Database	Routing Table	• List of routes generated when an algorithm is run on the link-state database. • The routing table of each router is unique and contains information on how and where to send packets to other routers. • Can be viewed using the **show ip route** command.

OSPF Components/ Algorithm

- The router builds the topology table using results of calculations based on the Dijkstra shortest-path first (SPF) algorithm.
- The SPF algorithm is based on the cumulative cost to reach a destination.
- The SPF algorithm creates an SPF tree by placing each router at the root of the tree and calculating the shortest path to each node.
- The SPF tree is then used to calculate the best routes.
- OSPF places the best routes into the forwarding database, which is used to make the routing table.



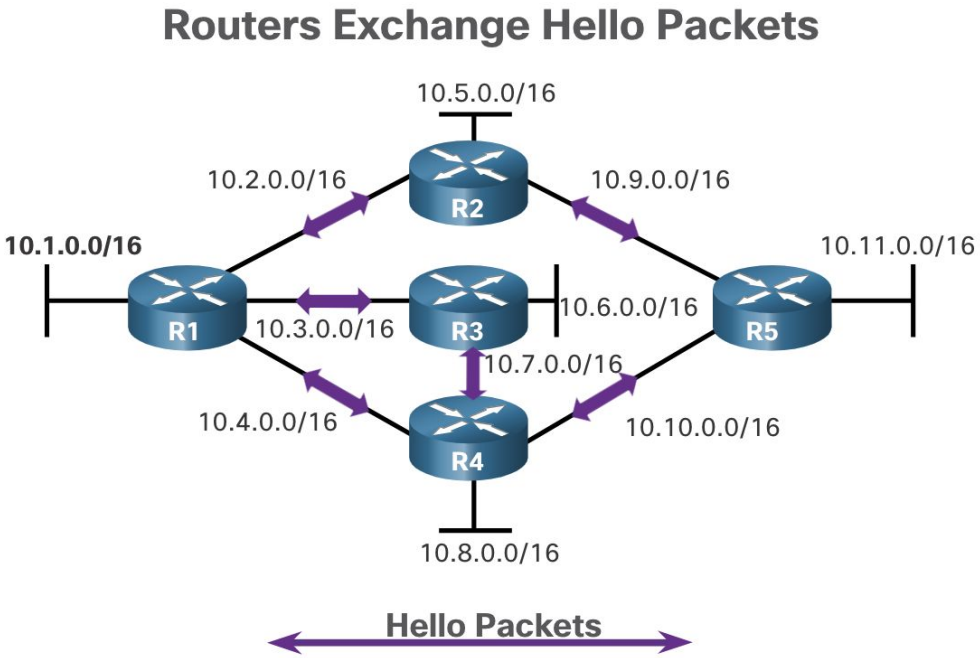
Link-State Operation

- *OSPF is a link-state routing protocol*
- To maintain routing information, OSPF routers complete a generic **link-state routing process** to reach a state of convergence.
- Each link between routers is labeled with a **cost value**.
- In OSPF, **cost** is used to determine **the best path** to the destination.
- The following are the link-state routing steps that are completed by a router:
  1. Establish Neighbor Adjacencies
  2. Exchange Link-State Advertisements
  3. Build the Link State Database
  4. Execute the SPF Algorithm
  5. Choose the Best Route

Link-State Operation/ Establishing Neighbor Adjacencies

1. Establish Neighbor Adjacencies

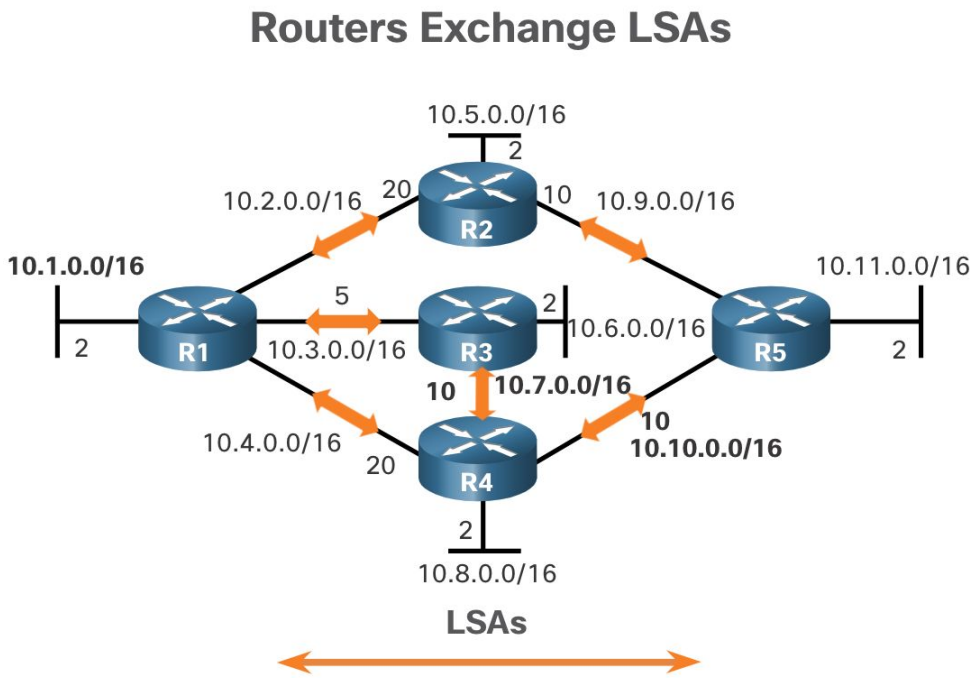
- OSPF-enabled routers must recognize each other on the network before they can share information.
- An OSPF-enabled router sends **Hello packets** out all OSPF-enabled interfaces to determine if neighbors are present on those links.
- If a neighbor is present, the OSPF-enabled router attempts to establish a neighbor adjacency with that neighbor.



Link-State Operation/ Exchange Link-State Advertisements

2. Exchange Link-State Advertisements

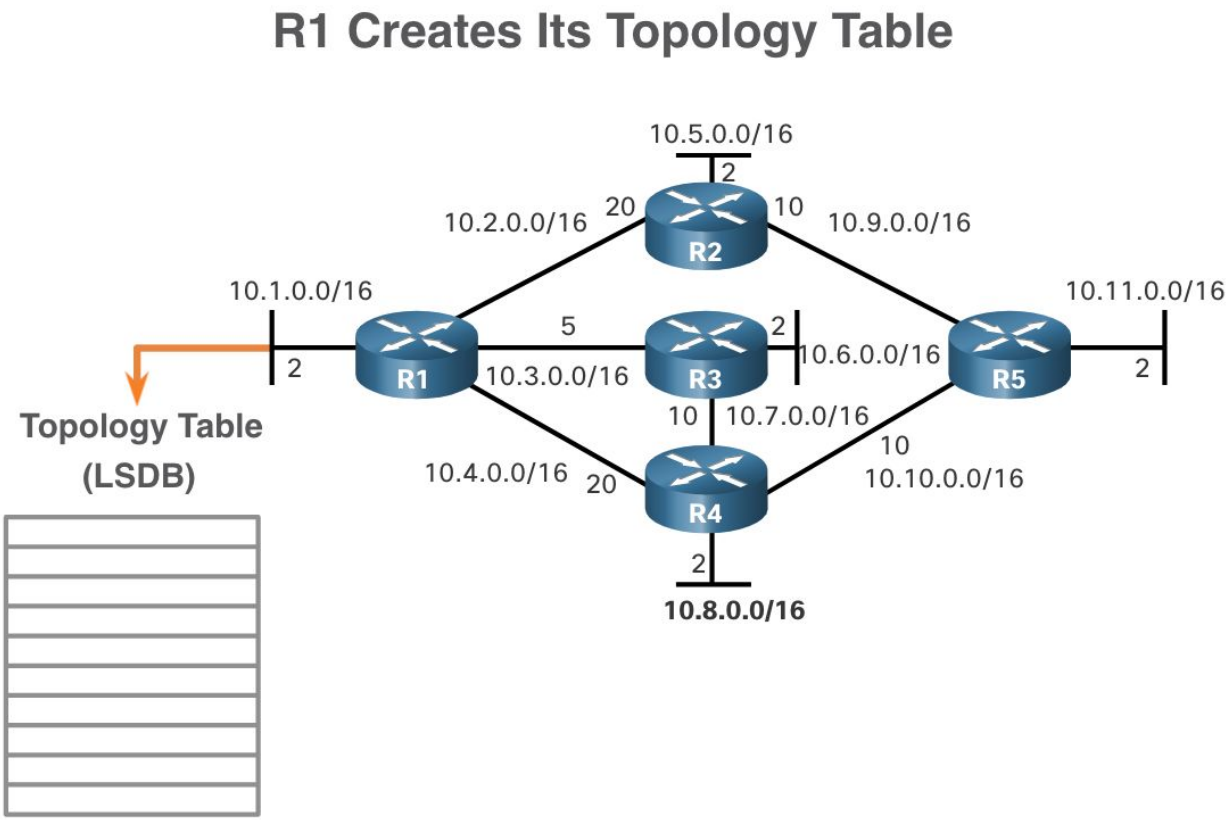
- After adjacencies are established, routers then exchange link-state advertisements (LSAs).
- LSAs contain the state and cost of each directly connected link.
- Routers flood their LSAs to adjacent neighbors.
- Adjacent neighbors receiving the LSA immediately flood the LSA to other directly connected neighbors, until all routers in the area have all LSAs.



Link-State Operation/ Build the Link State Database

3. Build the Link State Database

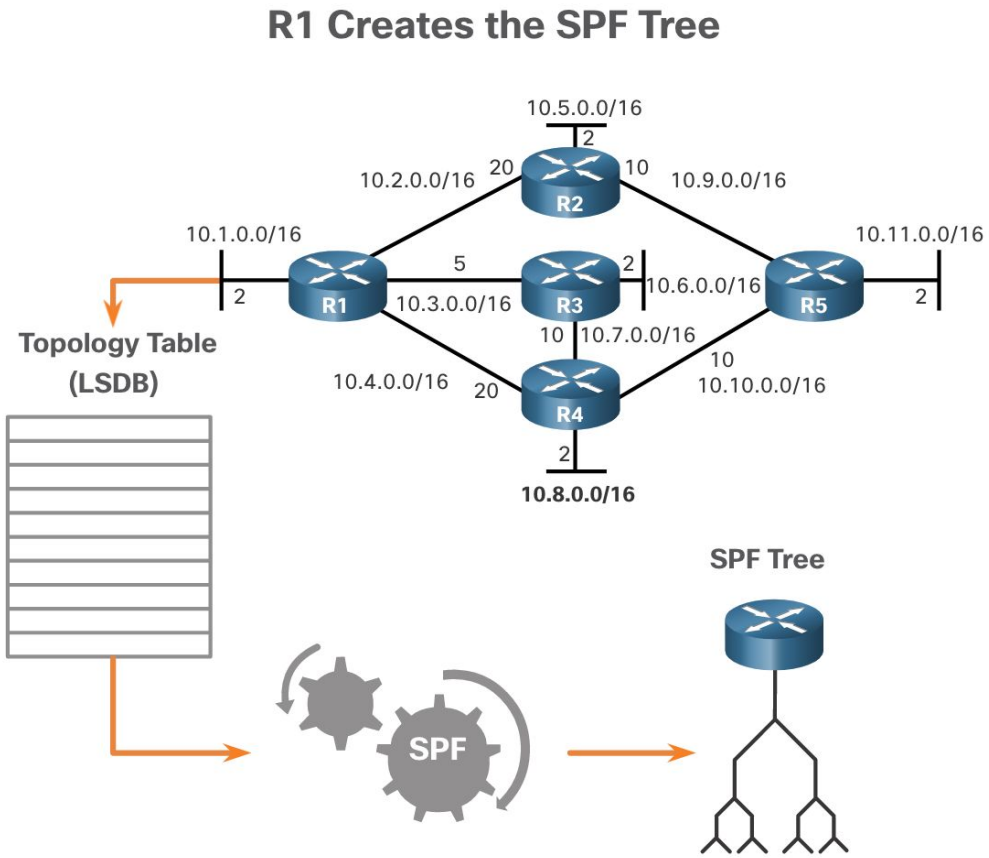
- After LSAs are received, OSPF-enabled routers build the topology table (LSDB).
- This database eventually holds all the information about the topology of the area.



Static Route/ Execute the SPF Algorithm

4. Execute the SPF Algorithm

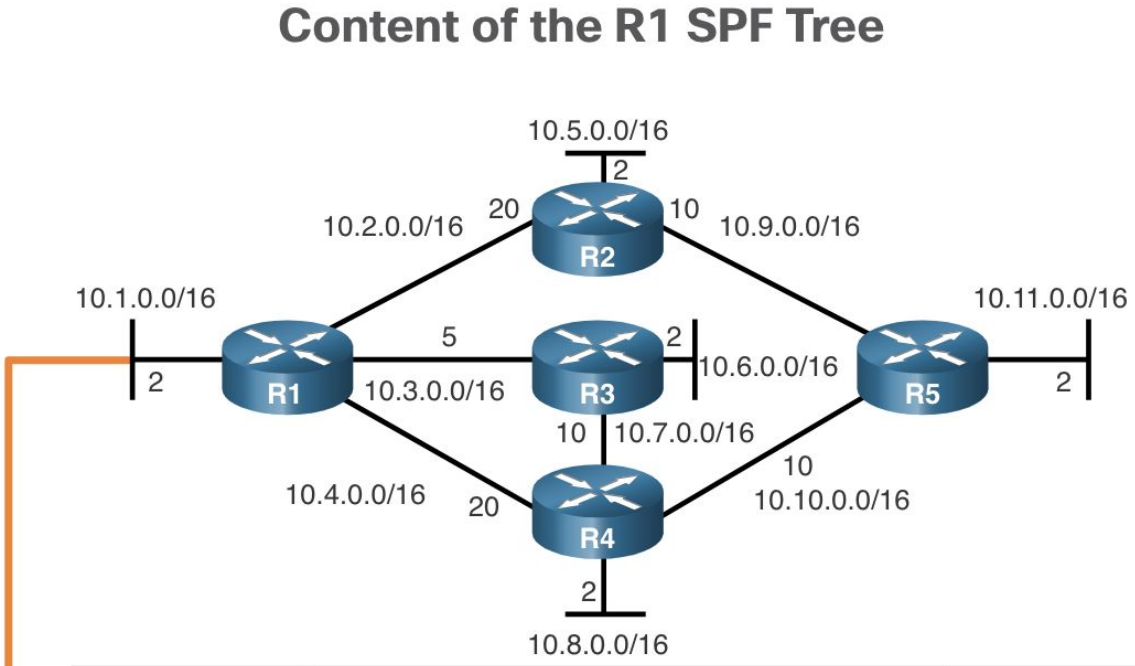
- Routers then execute the SPF algorithm.
- The SPF algorithm creates the SPF tree.



Static Route/ Choose the Best Route

5. Choose the Best Route

- After the SPF tree is built, the best paths to each network are offered to the IP routing table.
- The route will be inserted into the routing table unless there is a route source to the same network with a lower administrative distance, such as a static route.
- Routing decisions are made based on the entries in the routing table.



Destination	Shortest Path	Cost
10.5.0.0/16	R1→R2	22
10.6.0.0/16	R1→R3	7
10.7.0.0/16	R1→R3	15
10.8.0.0/16	R1→R3→R4	17
10.9.0.0/16	R1→R2	30
10.10.0.0/16	R1→R3→R4	25
10.11.0.0/16	R1→R3→R4→R5	27

Single-Area and Multiarea OSPF

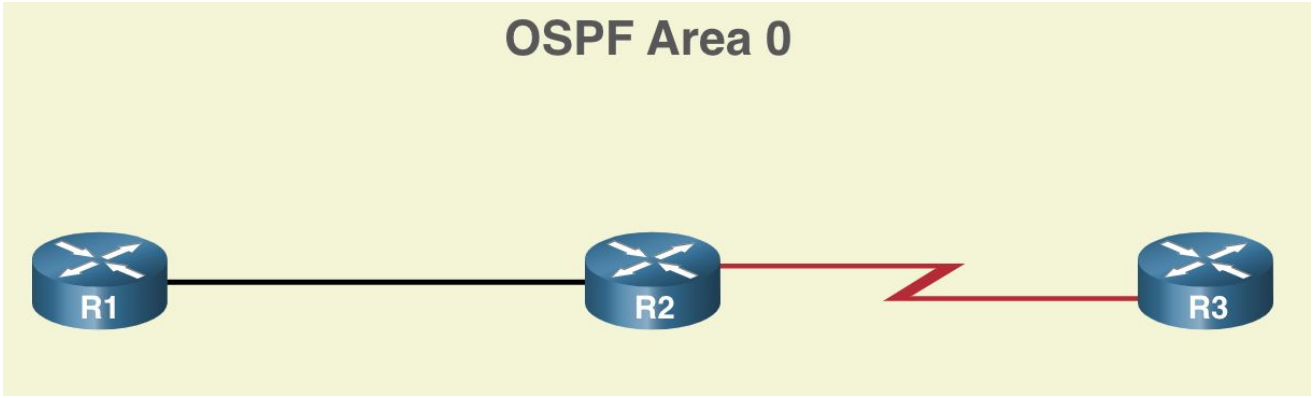
To make OSPF more efficient and scalable, OSPF supports hierarchical routing using areas.

An OSPF area is a group of routers that share the same link-state information in their LSDBs.

OSPF can be implemented in one of two ways, as follows:

- Single-Area OSPF - All routers are in one area. Best practice is to use area 0.
- Multiarea OSPF - OSPF is implemented using multiple areas, in a hierarchical fashion. All areas must connect to the backbone area (area 0). Routers interconnecting the areas are referred to as Area Border Routers (ABRs).

Single-Area OSPF

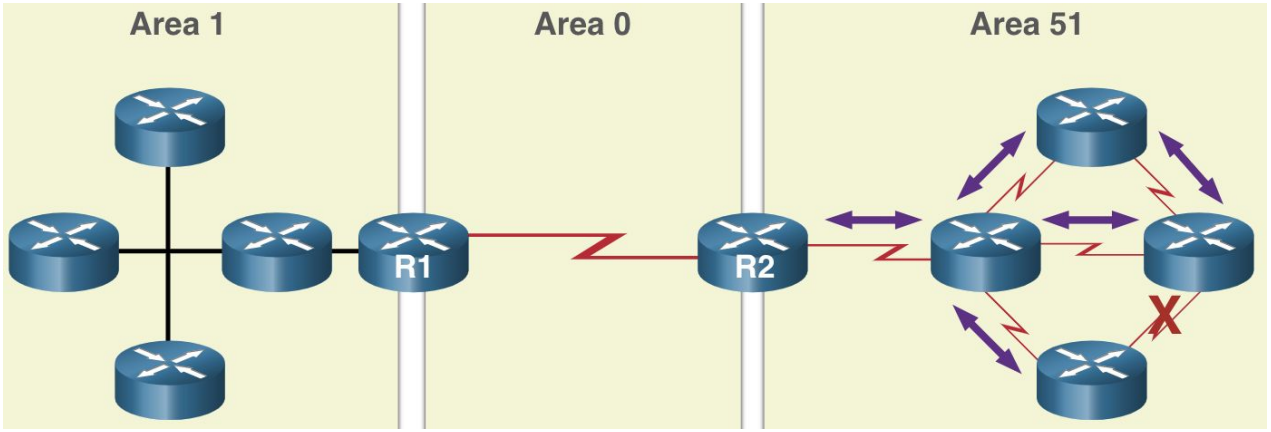


Multiarea OSPF

- One large routing domain can be divided into smaller areas, to support hierarchical routing.
- Routing still occurs between the areas (interarea routing).
- Routing operations, such as recalculating the database, are kept within an area.
- For instance, any time a router receives new information about a topology change within the area (including the addition, deletion, or modification of a link) the router must rerun the SPF algorithm, create a new SPF tree, and update the routing table.
- The SPF algorithm is CPU-intensive and the time it takes for calculation depends on the size of the area.
- Note: Routers in other areas receive updates regarding topology changes, but these routers only update the routing table, not rerun the SPF algorithm.

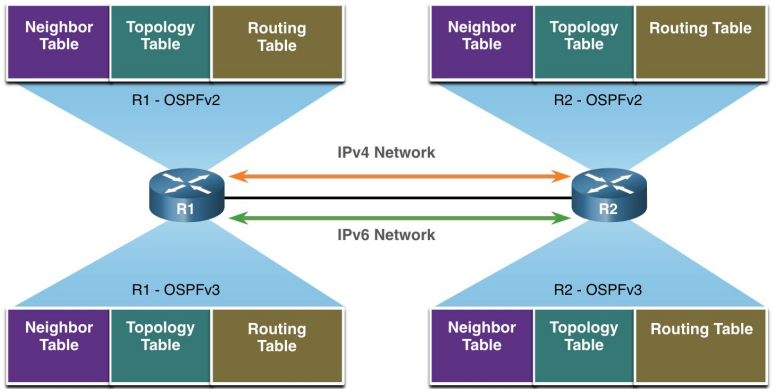
- Link failure affects the local area only (area 51).
- The ABR (R2) isolates the flooding of a specific LSA to area 51.
- Routers in areas 0 and 1 do not need to run the SPF algorithm.

Multi-Area OSPF



OSPFv3

- OSPFv3 is the OSPFv2 equivalent for exchanging IPv6 prefixes.
- Similar to its IPv4 counterpart, OSPFv3 exchanges routing information to populate the IPv6 routing table with remote prefixes.
- OSPFv3 Address Families feature, OSPFv3 includes support for both IPv4 and IPv6.
- OSPFv3 also uses the SPF algorithm as the computation engine to determine the best paths throughout the routing domain.
- OSPFv3 has separate processes from its IPv4 counterpart.
- The processes and operations are the same as in the IPv4 routing protocol, but run independently.
- OSPFv2 and OSPFv3 each have separate adjacency tables, OSPF topology tables, and IP routing tables.
- The OSPFv3 configuration and verification commands are similar to those used in OSPFv2.



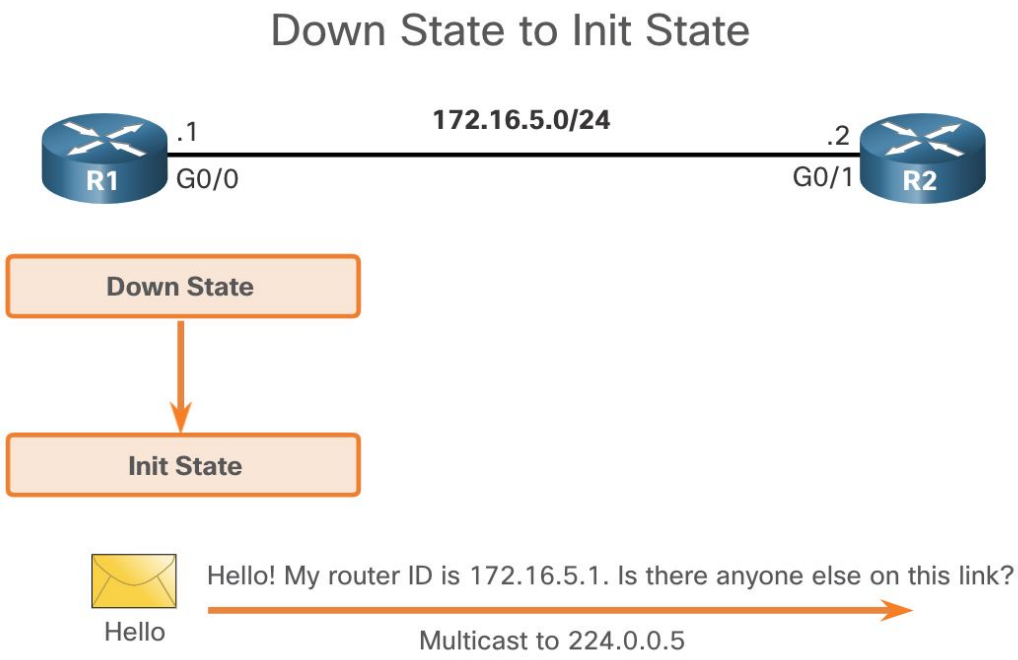
Types of OSPF Packets

- Link-state packets are the tools used by OSPF to help determine the fastest available route for a packet.
- OSPF uses the following link-state packets (LSPs) to establish and maintain neighbor adjacencies and exchange routing updates.
- Each packet serves a specific purpose in the OSPF routing process, as follows:

Type	Packet Name	Description
1	Hello	Discovers neighbors and builds adjacencies between them
2	Database Description (DBD)	Checks for database synchronization between routers
3	Link-State Request (LSR)	Requests specific link-state records from router to router
4	Link-State Update (LSU)	Sends specifically requested link-state records
5	Link-State Acknowledgment (LSAck)	Acknowledges the other packet types

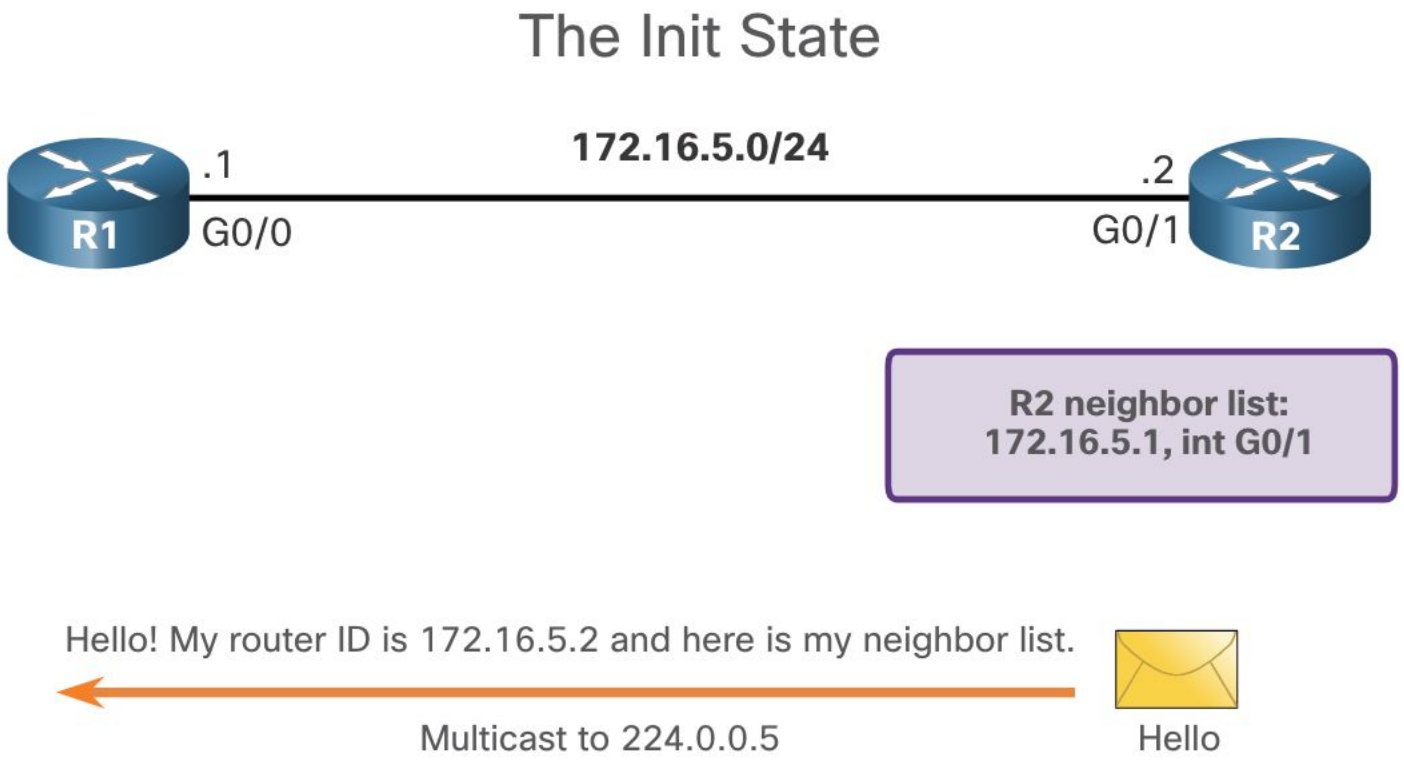
Establishing Neighbor Adjacencies

- When OSPF is enabled on an interface, the router must determine if there is another OSPF neighbor on the link by sending a **Hello packet** that contains its **router ID** out all OSPF-enabled interfaces.
- The Hello packet is sent to the reserved all OSPF Routers IPv4 multicast address 224.0.0.5.
- The OSPF router ID is a a 32-bit number that uniquely identifies each router in the OSPF area.
- When a neighboring OSPF-enabled router receives a **Hello packet** with a **router ID** that is not within its neighbor list, the receiving router attempts to establish an adjacency with the initiating router.



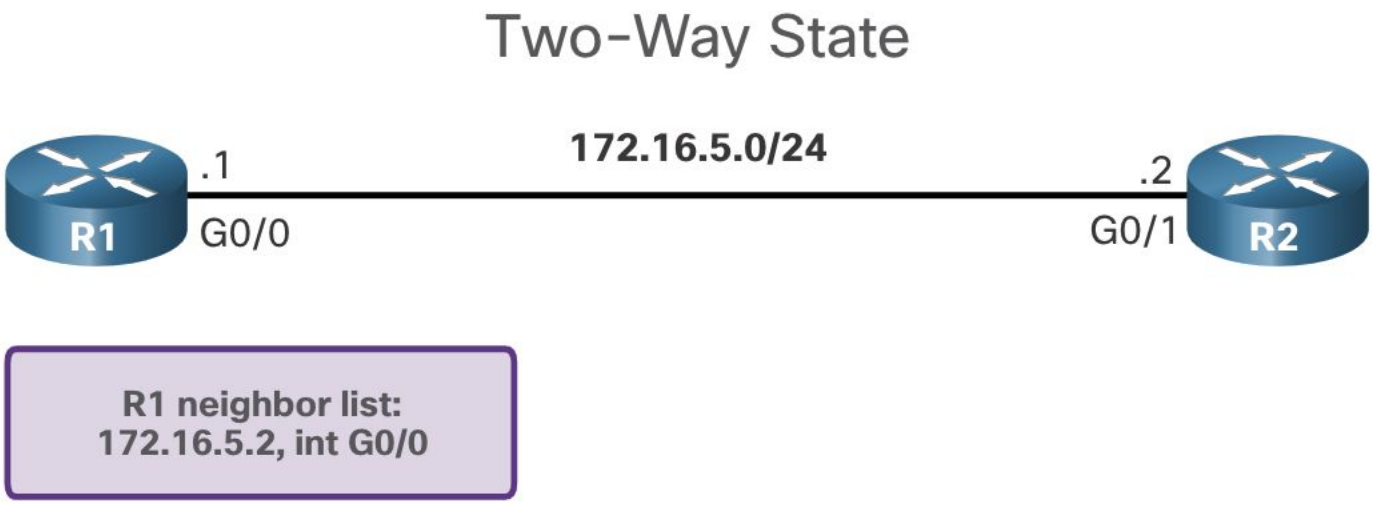
Establishing Neighbor Adjacencies (cont.)

- R2 receives the Hello packet from R1 and adds the R1 router ID to its neighbor list.
- R2 then sends a Hello packet to R1.
- The packet contains the R2 Router ID and the R1 Router ID in its list of neighbors on the same interface.



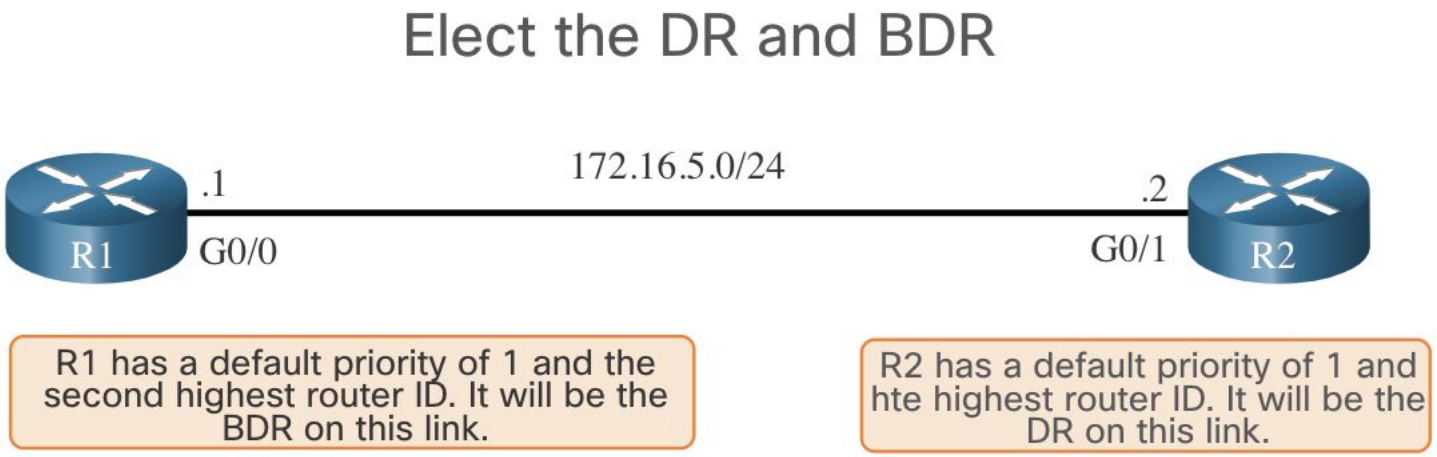
Establishing Neighbor Adjacencies (cont.)

- R1 receives the Hello and adds the R2 Router ID to its list of OSPF neighbors.
- It also notices its own Router ID in the list of neighbors of the Hello packet.
- When a router receives a Hello packet with its Router ID listed in the list of neighbors, the router transitions from the Init state to the Two-Way state.



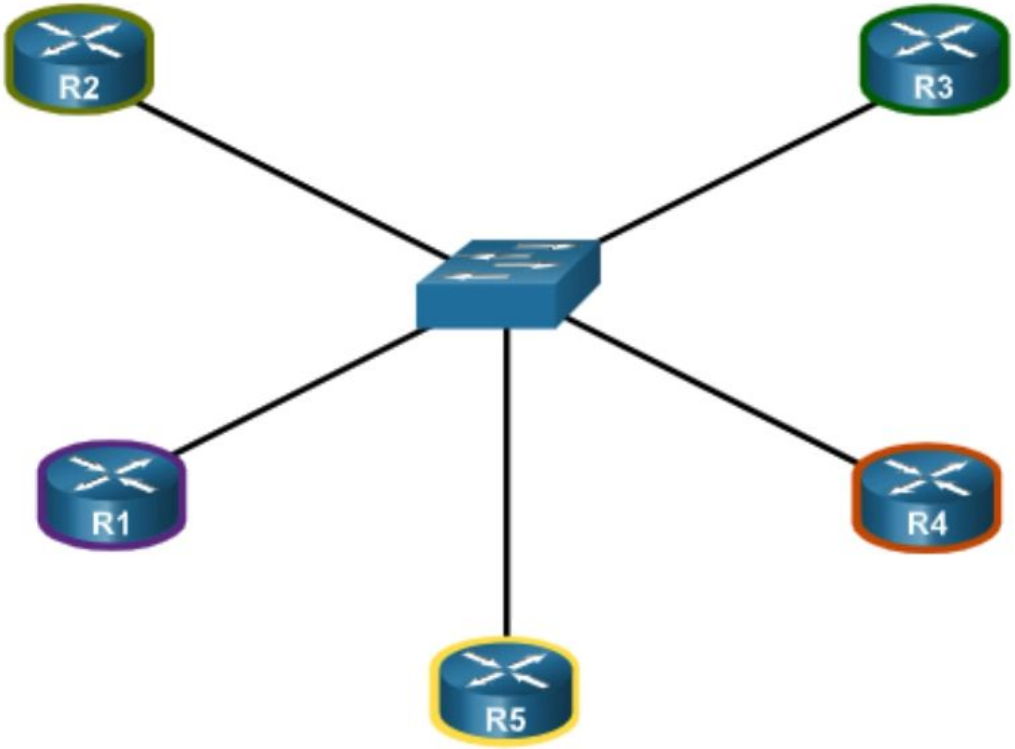
Establishing Neighbor Adjacencies (cont.)

- Because R1 and R2 are interconnected over an Ethernet network, a DR and BDR election takes place.
- This process only occurs on multiaccess networks such as Ethernet LANs.
- Hello packets are continually exchanged to maintain router information.
- As shown in the figure, R2 becomes the DR and R1 is the BDR.



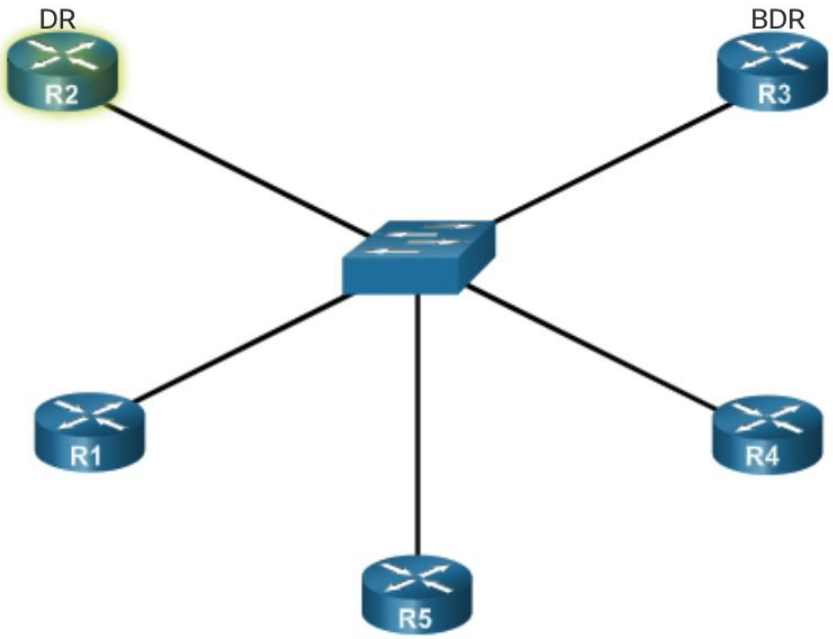
LSA Flooding Without a DR

- A dramatic increase in the number of routers also dramatically increases the number of LSAs exchanged between the routers.
- This flooding of LSAs significantly impacts the operation of OSPF.



LSA Flooding With a DR

- The solution to managing the number of adjacencies and the flooding of LSAs on a multiaccess network is the DR.
- On multiaccess networks, OSPF elects a DR to be the collection and distribution point for LSAs sent and received.
- A BDR is also elected in case the DR fails.



Router Configuration Mode for OSPF

- OSPFv2 is enabled using the {**router ospf** *process-id*} global configuration mode command.
- The *process-id* value represents a number between 1 and 65,535 and is selected by the network administrator.
- The *process-id* value is locally significant, which means that it does not have to be the same value on the other OSPF routers to establish adjacencies with those neighbors.
- It is considered best practice to use the same *process-id* on all OSPF routers.

```
R1(config)# router ospf 10
R1(config-router)# ?
  area                    OSPF area parameters
  auto-cost               Calculate OSPF interface cost according to bandwidth
  default-information     Control distribution of default information
  distance                Define an administrative distance
  exit                    Exit from routing protocol configuration mode
  log-adjacency-changes   Log changes in adjacency state
  neighbor                Specify a neighbor router
  network                 Enable routing on an IP network
  no                      Negate a command or set its defaults
  passive-interface       Suppress routing updates on an interface
  redistribute            Redistribute information from another routing protocol
  router-id               router-id for this OSPF process
R1(config-router)#
```

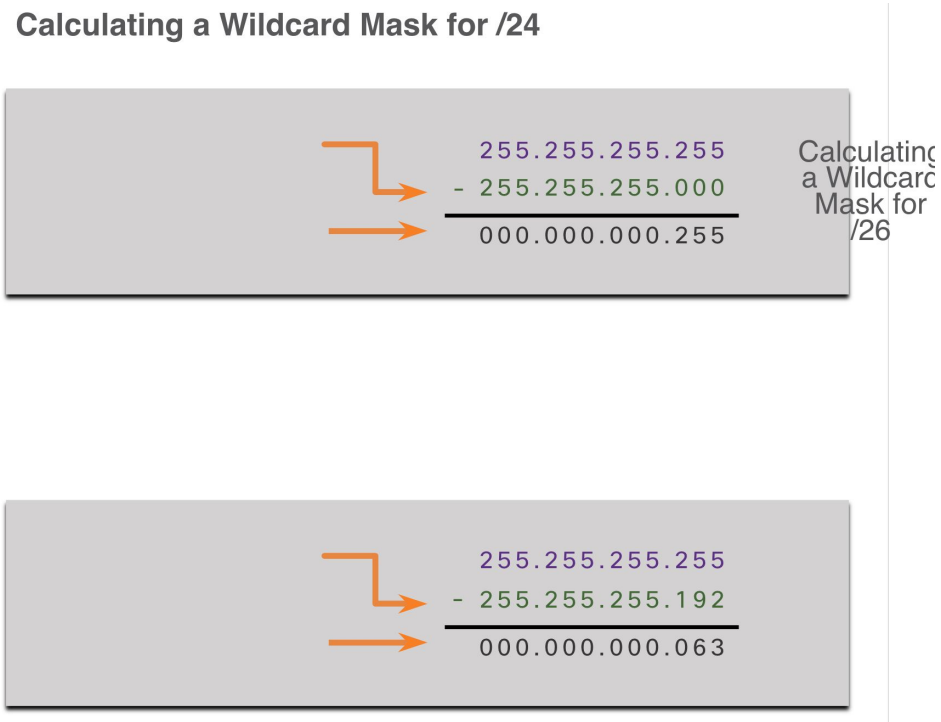
Router Configuration Mode for OSPF

- The *network-address wildcard-mask* syntax is used to enable OSPF on interfaces. Any interfaces on a router that match the network address in the **network** command are enabled to send and receive OSPF packets.
- The **area** *area-id* syntax refers to the OSPF area.
- When configuring single-area OSPFv2, the **network** command **must be** configured with the same *area-id* value on all routers.
- It is good practice to use an area ID of 0 with single-area OSPFv2.

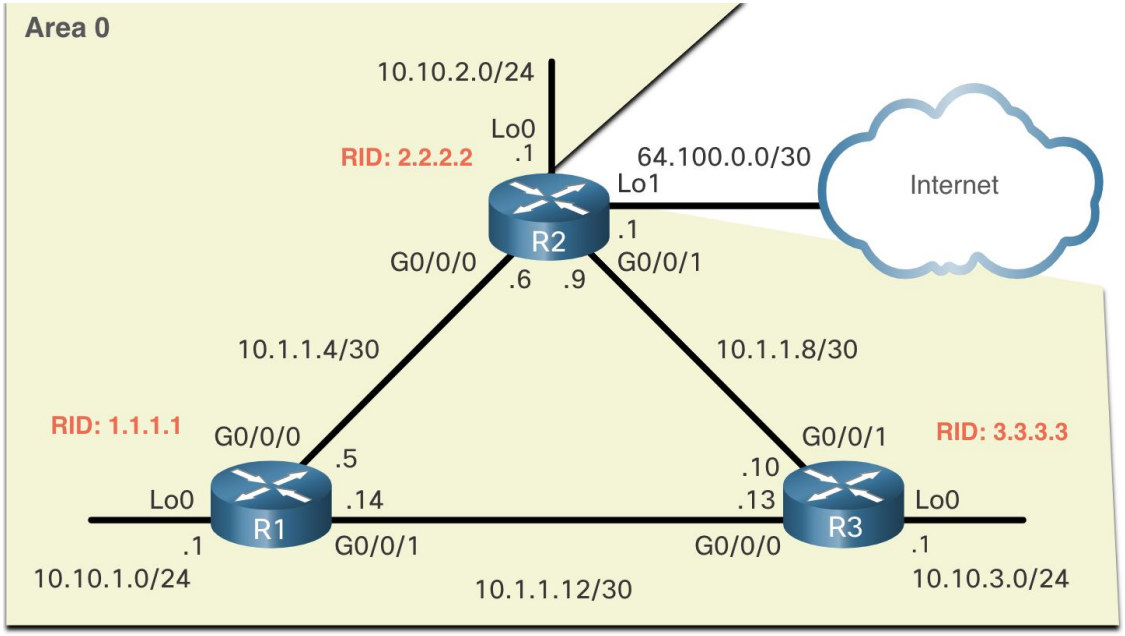
```
Router(config-router)# network network-address wildcard-mask area area-id
```

The Wildcard Mask

- The wildcard mask is typically the inverse of the subnet mask configured on that interface.
- In a subnet mask, binary 1 is equal to a match and binary 0 is not a match.
- In a wildcard mask, the reverse is true.
  - Wildcard mask bit 0 - Matches the corresponding bit value in the address.
  - Wildcard mask bit 1 - Ignores the corresponding bit value in the address.
- The easiest method for calculating a wildcard mask is to subtract the network subnet mask from 255.255.255.255, as shown for /24 and /26 subnet masks in the figure.



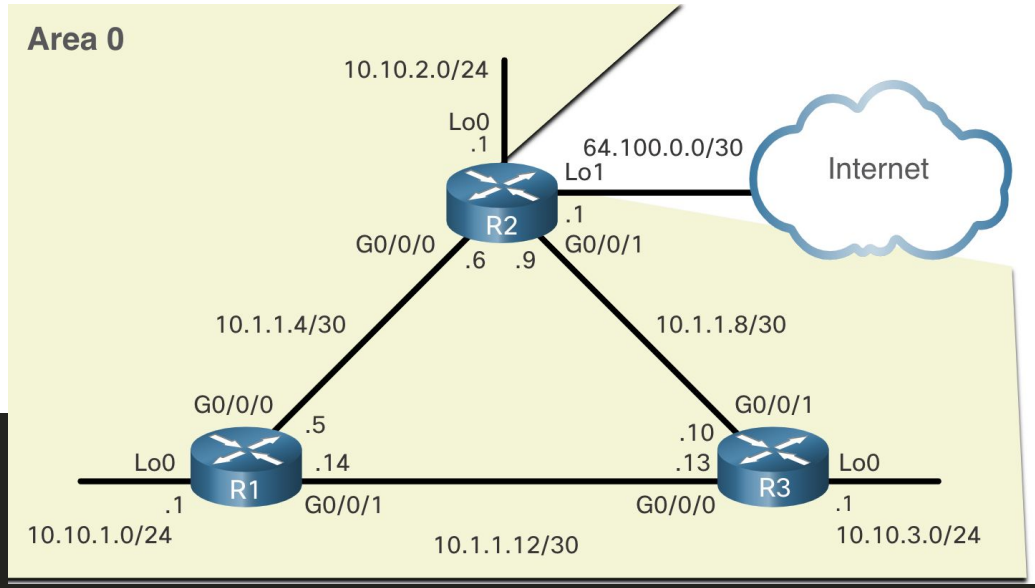
Configure a Router ID



- Use the **router-id** *rid* router configuration mode command to manually assign a router ID.
- In the example, the router ID 1.1.1.1 is assigned to R1.
- Use the **show ip protocols** command to verify the router ID.

```
R1(config)# router ospf 10
R1(config-router)# router-id 1.1.1.1
R1(config-router)# end
*May 23 19:33:42.689: %SYS-5-CONFIG_I: Configured from console by console
R1# show ip protocols | include Router ID
Router ID 1.1.1.1
R1#
```

OSPF Configuration




```
R1(config)# router ospf 10
R1(config-router)# network 10.10.1.0 0.0.0.255 area 0
R1(config-router)# network 10.1.1.4 0.0.0.3 area 0
R1(config-router)# network 10.1.1.12 0.0.0.3 area 0
R1(config-router)#
```

```
R2(config-router)# network 10.10.2.0 0.0.0.255 area 0
R2(config-router)# network 10.1.1.4 0.0.0.3 area 0
R2(config-router)# network 10.1.1.8 0.0.0.3 area 0
R2(config-router)#
```

```
R3(config-router)# network 10.10.3.1 0.0.0.0 area 0
R3(config-router)# network 10.1.1.10 0.0.0.0 area 0
R3(config-router)# network 10.1.1.13 0.0.0.0 area 0
R3(config-router)#
*Mar 26 14:00:55.183: %OSPF-5-ADJCHG: Process 10, Nbr 1.1.1.1 on GigabitEthernet0/0/0 from
LOADING to FULL, Loading Done
*Mar 26 14:00:55.243: %OSPF-5-ADJCHG: Process 10, Nbr 2.2.2.2 on GigabitEthernet0/0/1 from
LOADING to FULL, Loading Done
R3#
```

Passive Interface

- By default, OSPF messages are forwarded out all OSPF-enabled interfaces.
- These messages really only need to be sent out interfaces that are connecting to other OSPF-enabled routers.
- Sending out unneeded messages on a LAN affects the network in three ways, as follows:
    - Inefficient Use of Bandwidth
    - Inefficient Use of Resources
    - Increased Security Risk

COURSE: **NETWORK ADMINISTRATION AND MANAGEMENT**

COURSE INSTRUCTOR: **MIRALDA CUKA, PHD**

Lecture 8
Wireless Networks

Benefits of Wireless

- A Wireless LAN (WLAN) is a type of wireless network that is commonly used in homes, offices, and campus environments.
- Networks must support people who are on the move.
- People connect using computers, laptops, tablets, and smart phones.
- There are many different network infrastructures that provide network access, such as wired LANs, service provider networks, and cell phone networks.
- But it's the WLAN that makes mobility possible within the home and business environments.

Different Wireless Networks

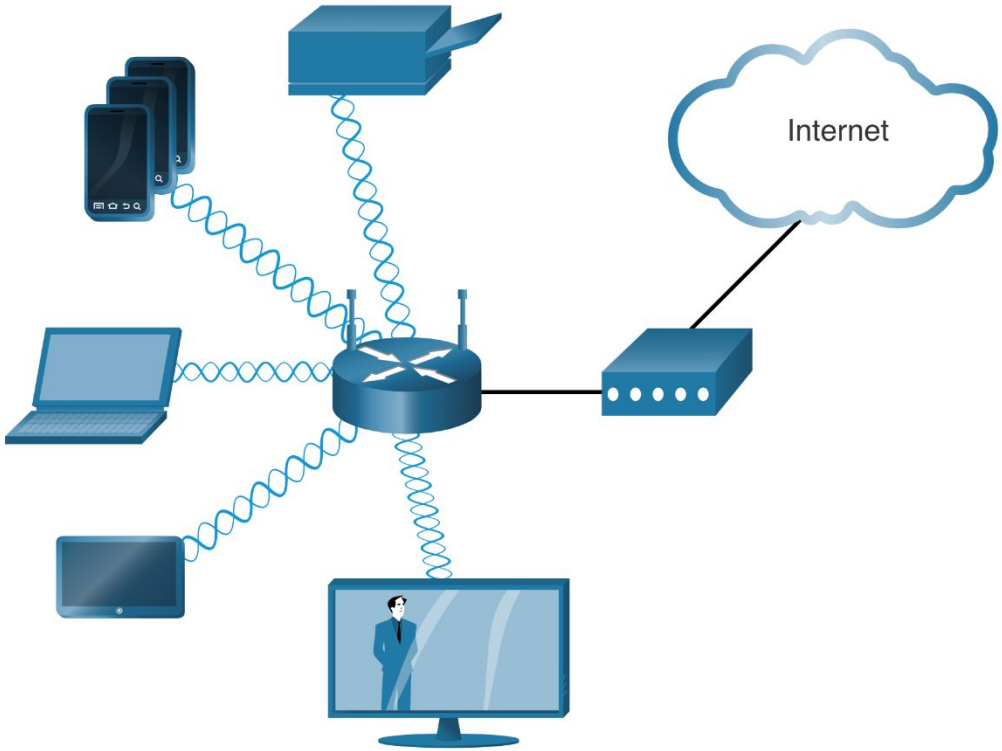
- Wireless networks are based on the Institute of Electrical and Electronics Engineers (IEEE) standards and can be classified broadly into four main types:
  - WPAN
  - WLAN
  - WMAN
  - WWAN

WPAN

- Wireless Personal-Area Networks (WPAN):
  - Use low powered transmitters for a short-range network (6-9m)
  - Bluetooth and ZigBee devices
  - Based on the 802.15 standard and a 2.4-GHz radio frequency

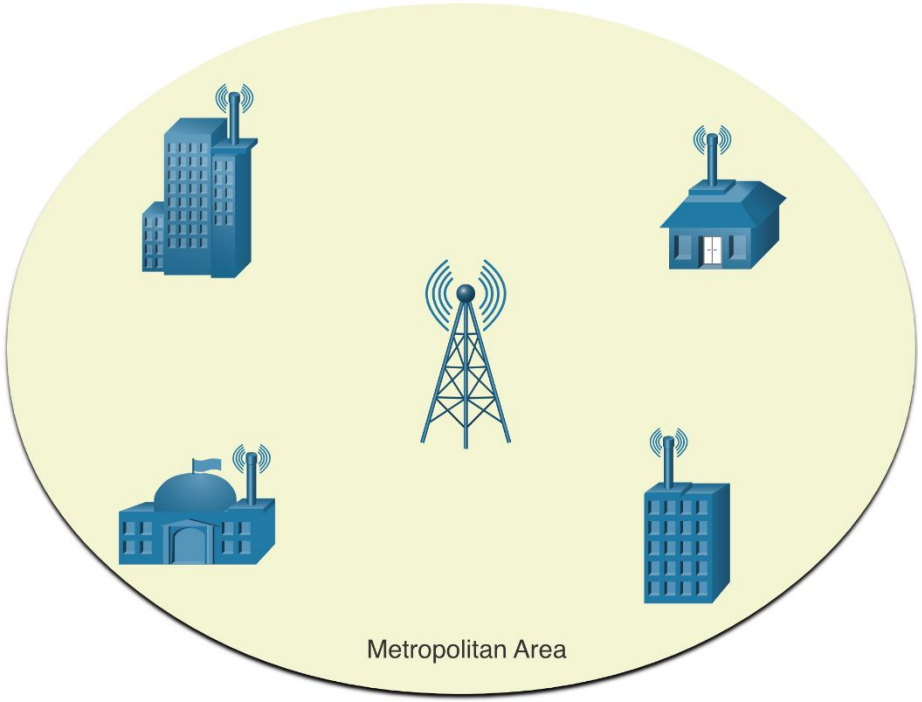
WLAN

- Wireless LANs (WLAN):
  - Uses transmitters to cover a medium-sized network (300m)
  - Suitable for use in a home, office, and even a campus environment
  - Based on the 802.11 standard and a 2.4-GHz or 5-GHz radio frequency



WMAN

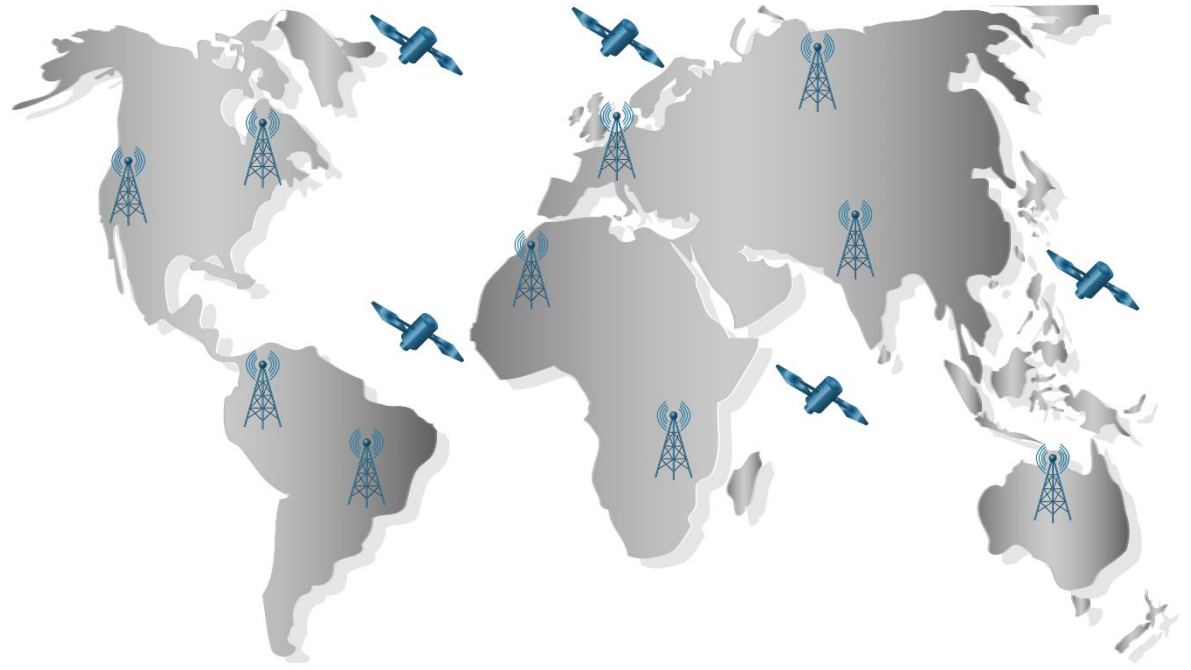
- Wireless MANs (WMAN):
  - Uses transmitters to provide wireless service over a larger geographic area.
  - WMANs are suitable for providing wireless access to a metropolitan city or specific district.
  - WMANs use specific licensed frequencies.



WWAN

Wireless Wide-Area Networks (WWANs):

- Uses transmitters to provide coverage over an extensive geographic area.
- WWANs are suitable for national and global communications. WWANs also use specific licensed frequencies.



Wireless Technologies

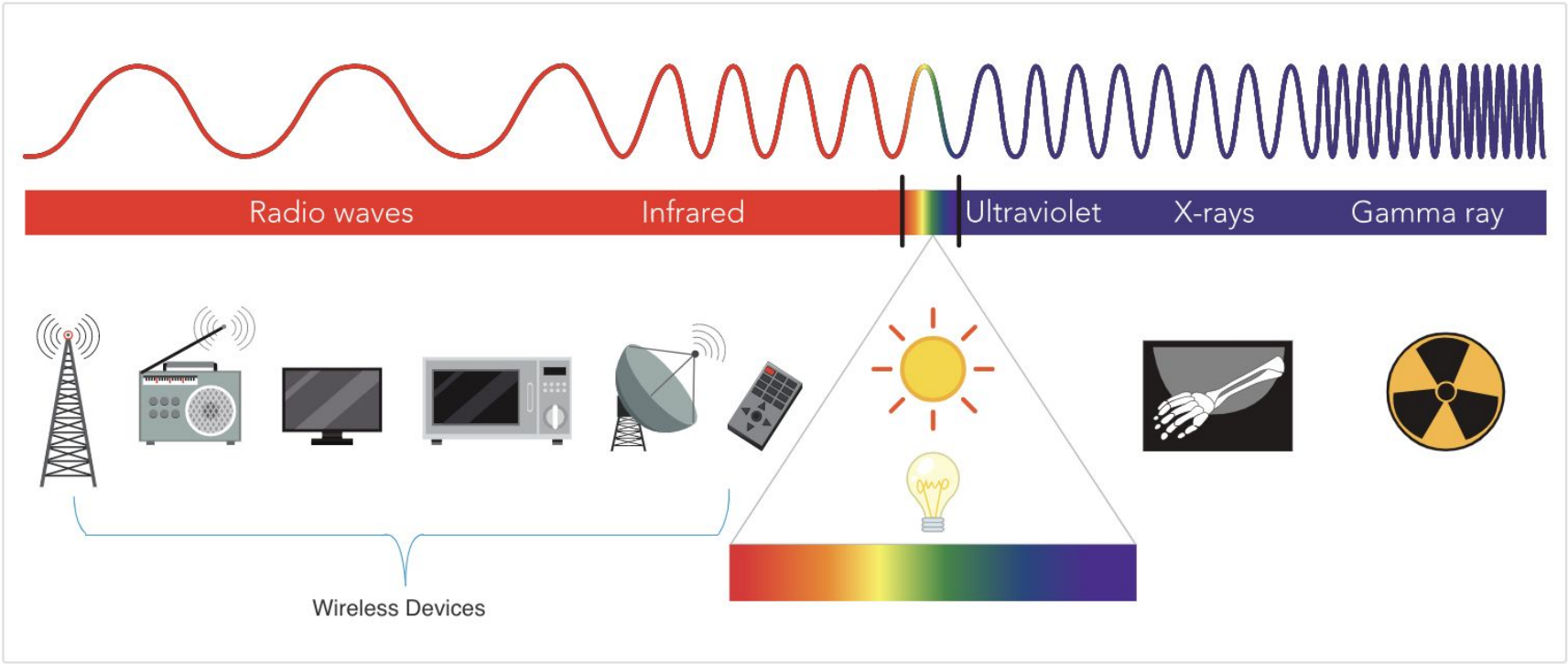
- Wireless technology uses the unlicensed radio spectrum to send and receive data.
- The unlicensed spectrum is accessible to anyone who has a wireless router and wireless technology in the device they are using.
- Wireless technology used:
  - Bluetooth
  - WiMAX
  - Cellular Broadband
  - Satellite Broadband

Wireless Standards

IEEE WLAN Standard	Radio Frequency	Description
802.11	2.4 GHz	• speeds of up to 2 Mbps
802.11a	5 GHz	• speeds of up to 54 Mbps • small coverage area • less effective at penetrating building structures • not interoperable with the 802.11b and 802.11g
802.11b	2.4 GHz	• speeds of up to 11 Mbps • longer range than 802.11a • better able to penetrate building structures
802.11g	2.4 GHz	• speeds of up to 54 Mbps • backward compatible with 802.11b with reduced bandwidth capacity
802.11n	2.4 GHz 5 GHz	• data rates range from 150 Mbps to 600 Mbps with a distance range of up to 70 m (230 feet) • APs and wireless clients require multiple antennas using MIMO technology • backward compatible with 802.11a/b/g devices with limiting data rates
802.11ac	5 GHz	• provides data rates ranging from 450 Mbps to 1.3 Gbps (1300 Mbps) using MIMO technology • Up to eight antennas can be supported • backwards compatible with 802.11a/n devices with limiting data rates
802.11ax	2.4 GHz 5 GHz	• latest standard released in 2019 • also known as Wi-Fi 6 or High-Efficiency Wireless (HEW) • provides improved power efficiency, higher data rates, increased capacity, and handles many connected devices • currently operates using 2.4 GHz and 5 GHz but will use 1 GHz and 7 GHz when those frequencies become available • Search the internet for Wi-Fi Generation 6 for more information

Radio Frequencies

- All wireless devices operate in the **radio waves range** of the electromagnetic spectrum.
- WLAN networks operate in the **2.4 GHz** frequency band and the **5 GHz band**.
- The following frequency bands are allocated to 802.11 wireless LANs:
  - 2.4 GHz (UHF) - 802.11b/g/n/ax
  - 5 GHz (SHF) - 802.11a/n/ac/ax



WLAN Components/ Wireless NICs

Wireless deployments require a minimum of two devices that have a **radio transmitter** and a **radio receiver** tuned to the same radio frequencies:

- End devices with wireless NICs
- A network device, such as a wireless router or wireless AP

To communicate wirelessly, devices must include integrated wireless NICs that incorporate a radio transmitter/receiver.

If a device does not have an integrated wireless NIC, then a USB wireless adapter can be used.

Wireless Home Router

- A home user typically interconnects wireless devices using a small, wireless router.
- The wireless router serves as an:
  - **Access point** - This provides 802.11a/b/g/n/ac wireless access.
  - **Switch** - This provides a four-port, full-duplex, 10/100/1000 Ethernet switch to interconnect wired devices.
  - **Router** - This provides a default gateway for connecting to other network infrastructures, such as the internet.

Wireless Home Router (cont.)

- The wireless router advertises its wireless services by sending beacons containing its shared service set identifier (SSID).
- Devices wirelessly discover the SSID and attempt to associate and authenticate with it to access the local network and internet.
- Most wireless routers also provide advanced features, such as:
  - high-speed access
  - support for video streaming
  - IPv6 addressing
  - quality of service (QoS)
  - USB ports to connect printers or portable drives
- Additionally, home users who want to extend their network services can implement Wi-Fi range extenders.
- A device can connect wirelessly to the extender, which boosts its communications to be repeated to the wireless router.

Wireless Access Point

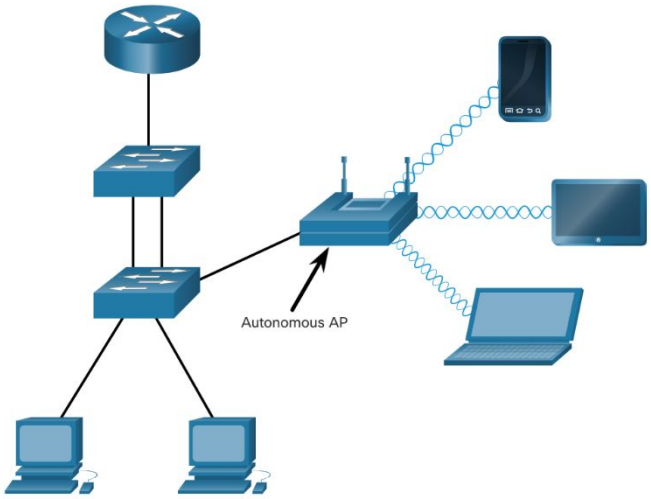
- An **access point** is a device that creates a wireless local area network, or WLAN, usually in an office or large building.
- An **access point** connects to a wired router, switch, or hub via an Ethernet cable, and projects a Wi-Fi signal to a designated area.

Note: Home Routers can be a AP, but an AP not always is a home router.

AP Categories

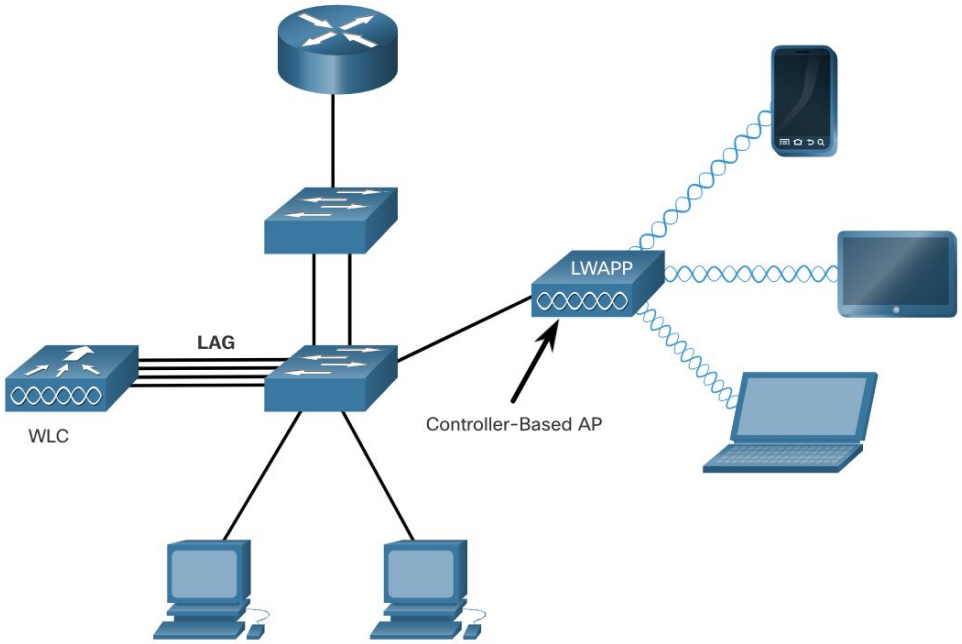
Autonomous APs

- These are standalone devices configured using a command line interface or a GUI. Autonomous APs are useful in situations where only a couple of APs are required in the organization.
- A home router is an example of an autonomous AP because the entire AP configuration resides on the device.
- If the wireless demands increase, more APs would be required.
- Each AP would operate independent of other APs and each AP would require manual configuration and management.



Controller-based APs

- These devices require no initial configuration and are often called lightweight APs (LAPs).
- LAPs use the Lightweight Access Point Protocol (LWAPP) to communicate with a WLAN controller (WLC).
- Controller-based APs are useful in situations where many APs are required in the network.
- As more APs are added, each AP is automatically configured and managed by the WLC.



Wireless Antennas

Omnidirectional Antennas

- Omnidirectional antennas provide 360-degree.

Directional Antennas

- Directional antennas focus the radio signal in a given direction.
- This enhances the signal to and from the AP in the direction the antenna is pointing
- This provides a stronger signal strength in one direction and reduced signal strength in all other directions.
- Examples of directional Wi-Fi antennas include Yagi and parabolic dish antennas.

MIMO Antennas

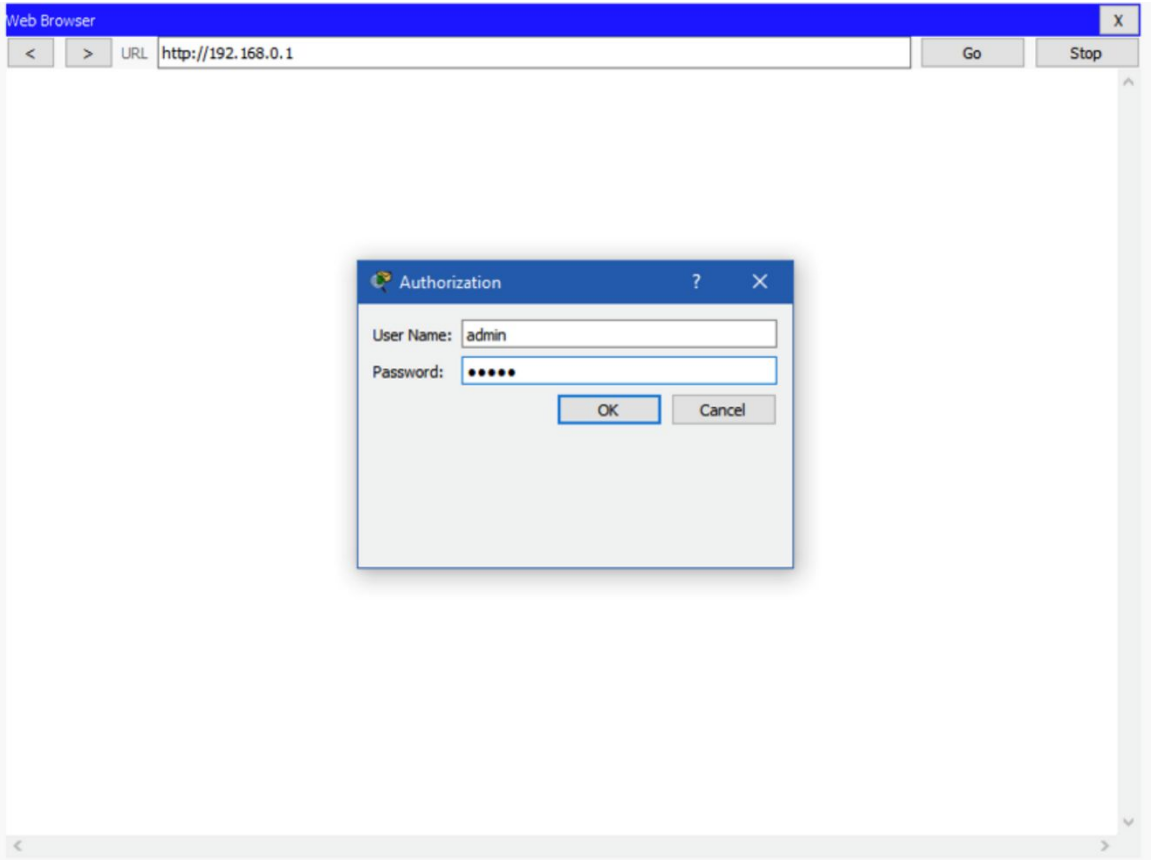
- Multiple Input Multiple Output (MIMO) uses multiple antennas.
- Increases available bandwidth for IEEE 802.11n/ac/ax wireless networks.
- Up to eight transmit and receive antennas can be used to increase throughput.

The Wireless Router

- Home networks often use a small office and home router.
- These routers are sometimes called an integrated router
- They typically include:
  - a switch for wired clients
  - a port for an internet connection (sometimes labeled “WAN”)
  - wireless components for wireless client access

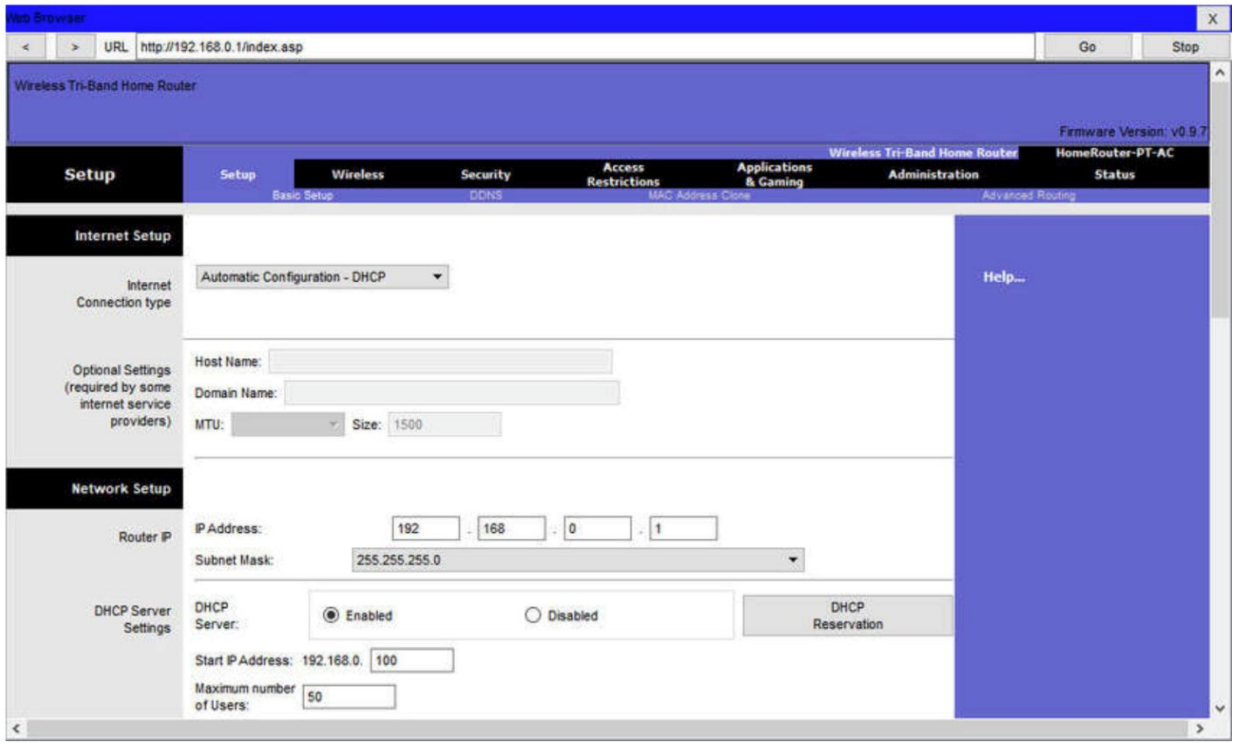
Step 1: Log in to the Wireless Router

- To gain access to the wireless router's configuration GUI, open a web browser.
- In the address field, enter the default IP address for your wireless router (usually 192.168.0.1 or 192.168.1.1)
- A security window prompts for authorization to access the router GUI.
- The word **admin** is commonly used as the default username and password.



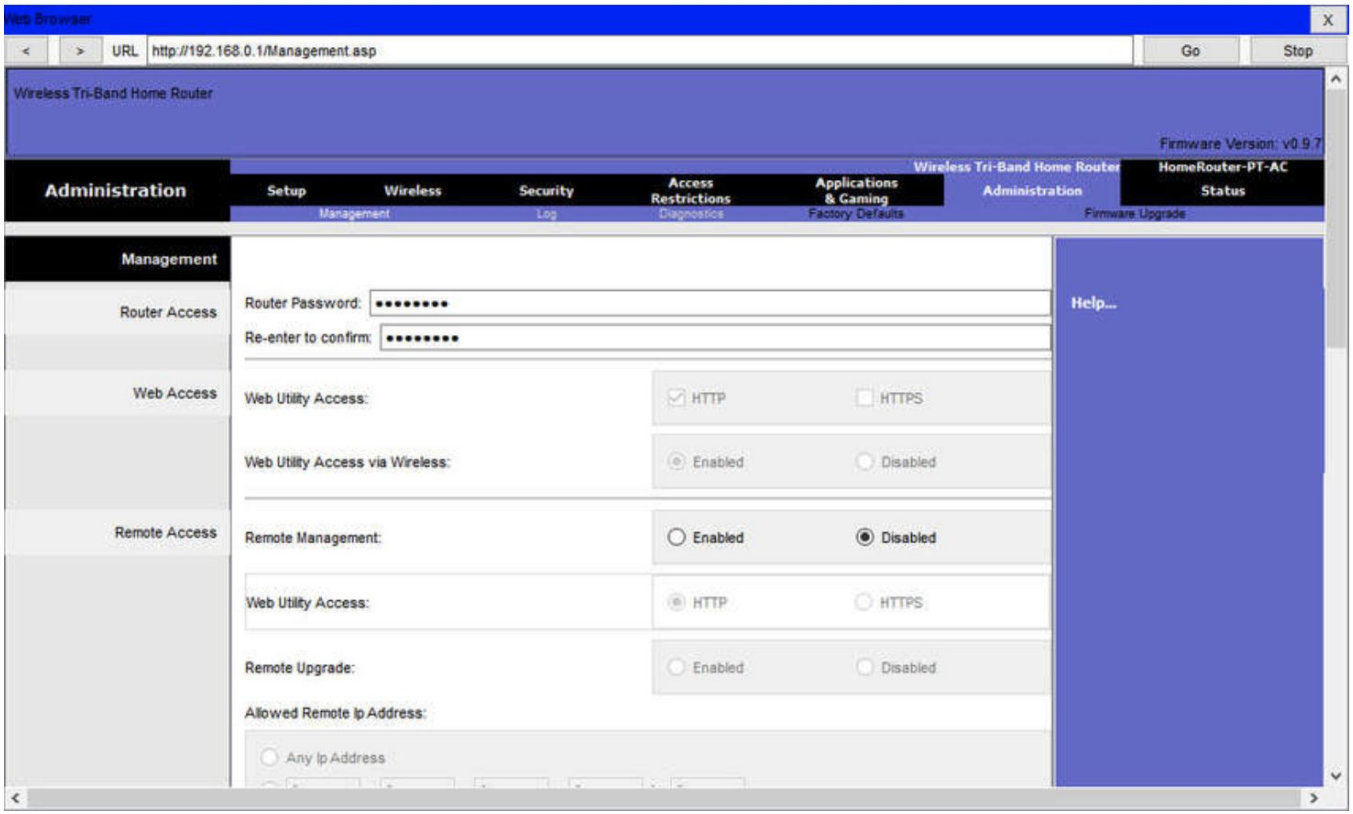
Step 1: Log in to the Wireless Router (cont.)

- After logging in, a GUI opens.
- The GUI will have tabs or menus to help you navigate to various router configuration tasks.
- It is often necessary to **save** the settings changed in one window **before proceeding** to another window.
- At this point, it is a best practice to make changes to the default settings.



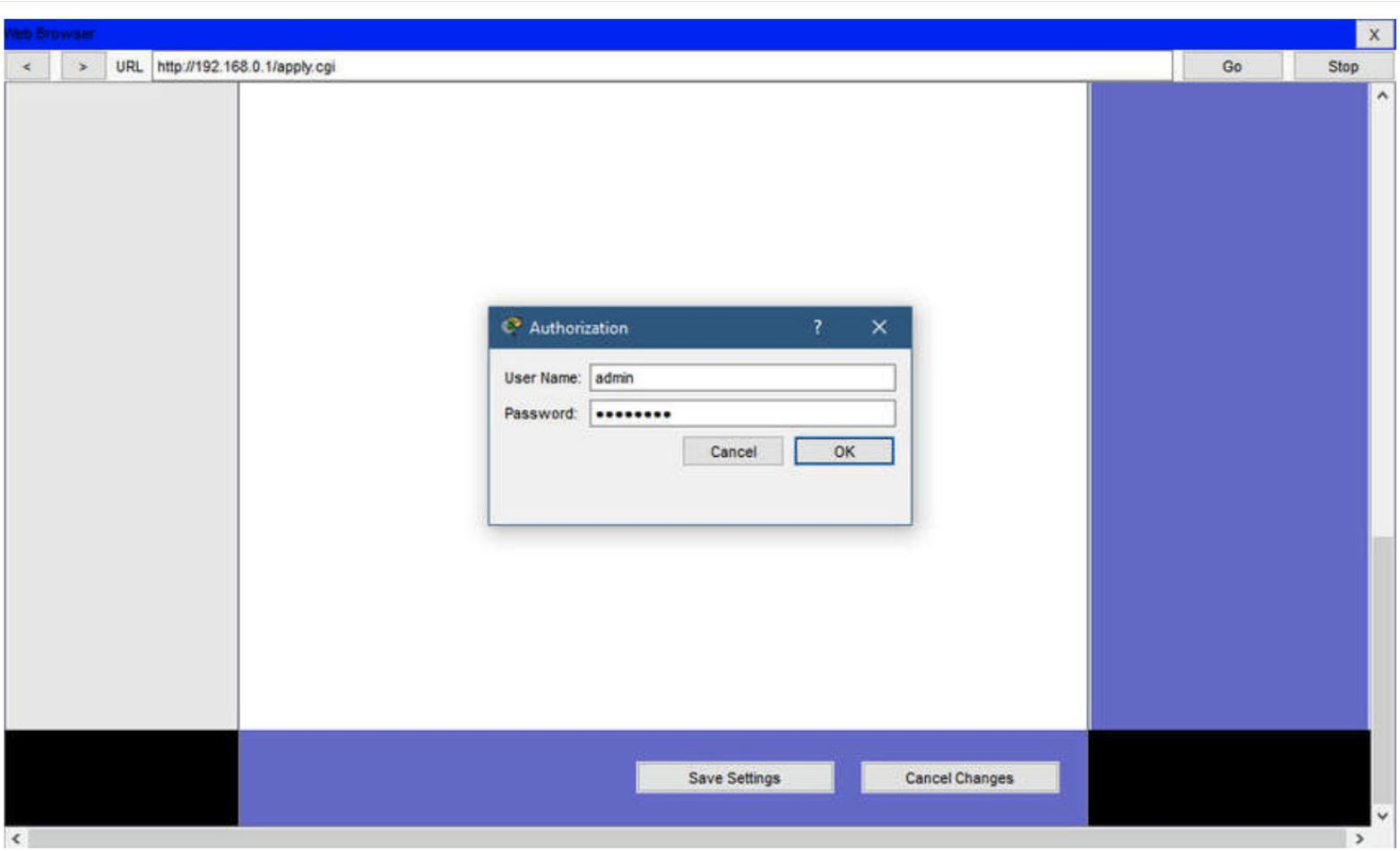
Step 2: Change the Default Administrative Password

- To change the default login password, find the administration portion of the router's GUI.
- In this example, the Administration tab was selected.
- This is where the router password can be changed.
- On some devices, such as the one in the example, you can only change the password.
- The username remains admin or whatever the default username is for the router you are configuring.



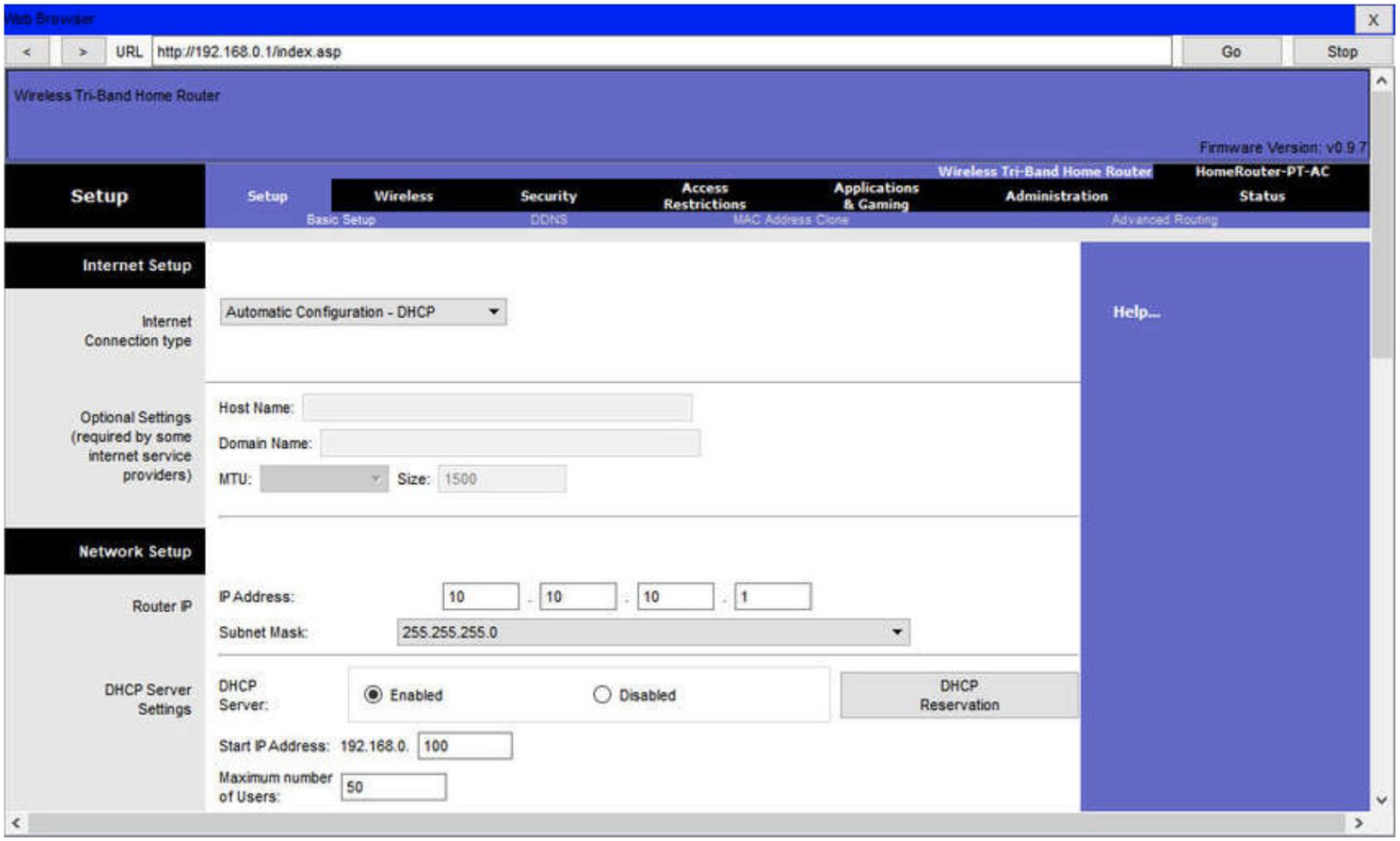
Step 3: Log in with the New Administrative Password

- After you save the new password, the wireless router will request authorization again.
- Enter the username and new password, as shown in the example.



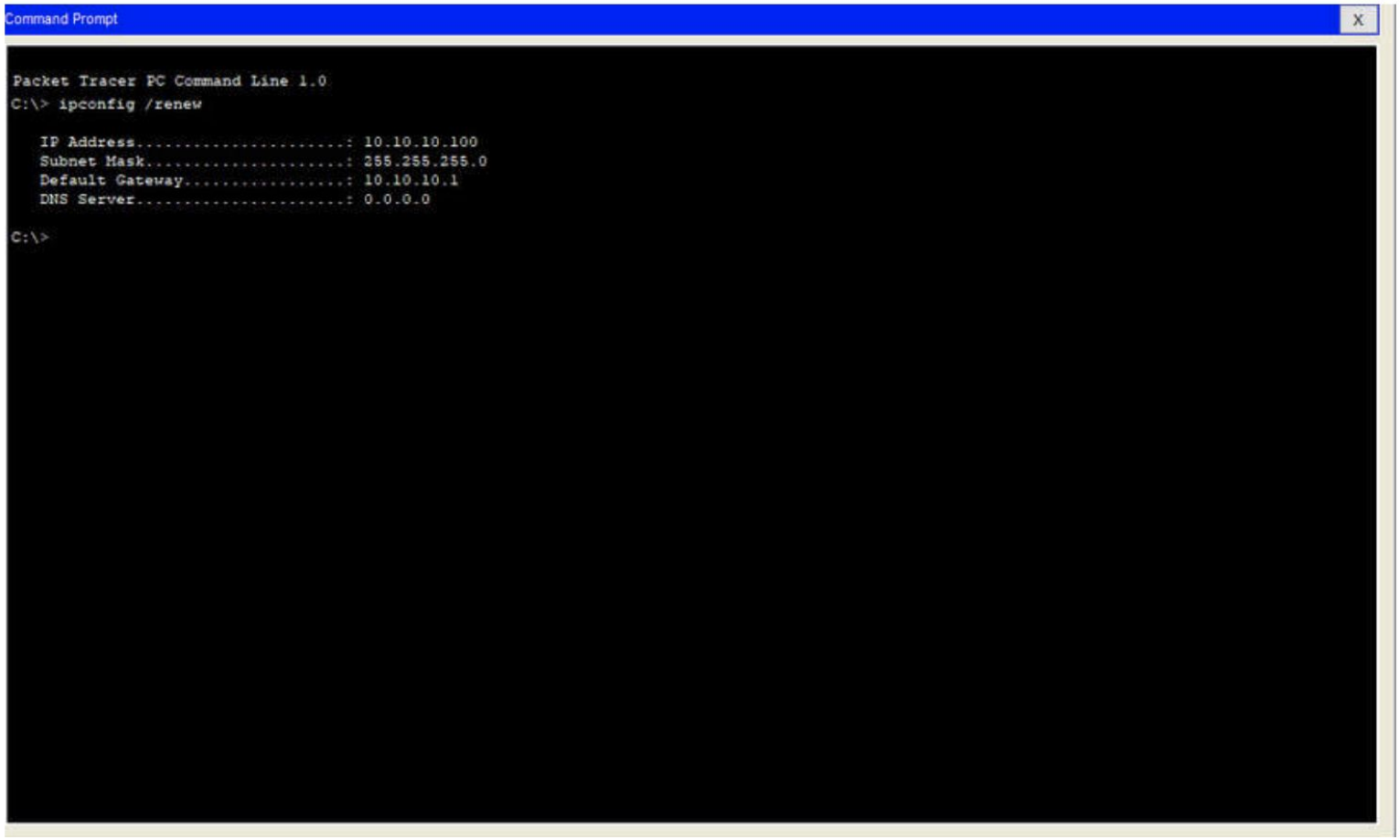
Step 4: Change the Default DHCP IPv4 Addresses

- Change the default router IPv4 address.
- It is a best practice to use private IPv4 addressing inside your network.
- The IPv4 address 10.10.10.1 is used in the example but it could be any private IPv4 address you choose.



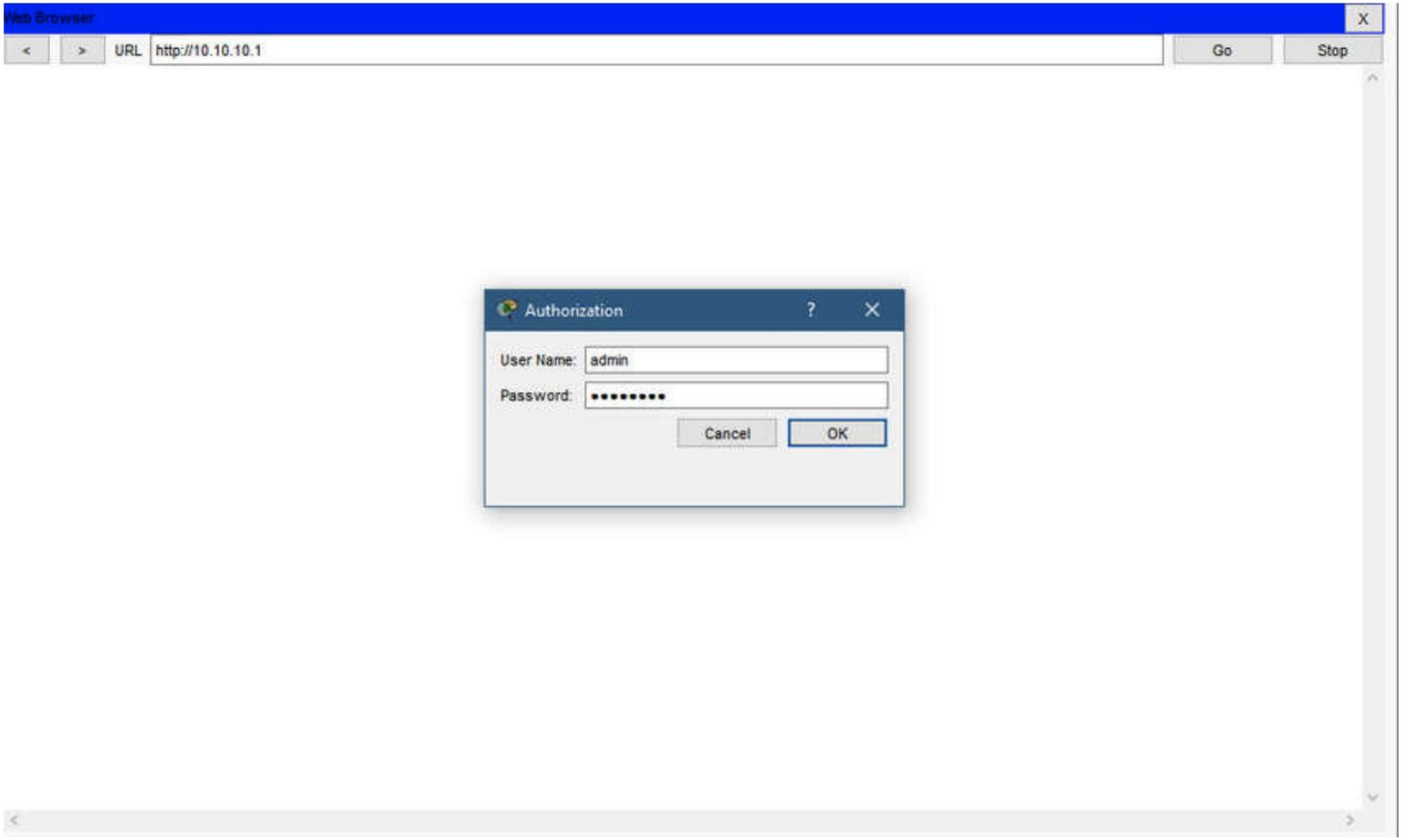
Step 5: Renew the IP Address

- When you click save, you will temporarily lose access to the wireless router.
- Open a command window and renew your IP address with the ipconfig /renew command, as shown in the example.



Step 6: Log in to the Router with the New IP Address

- Enter the router's new IP address to regain access to the router configuration GUI, as shown in the example.
- You are now ready to continue configuring the router for wireless access.



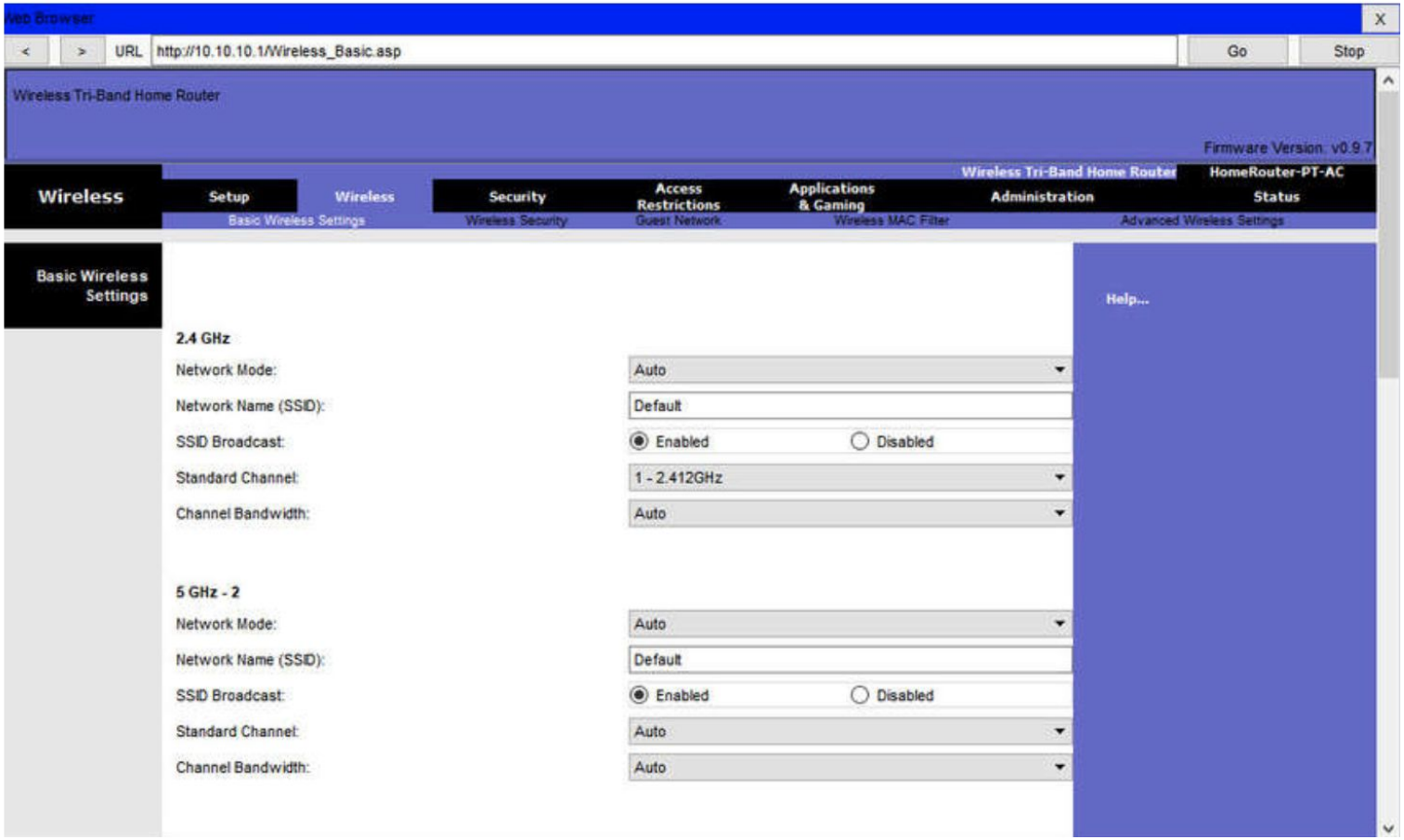
Basic Wireless Setup

Basic wireless setup includes the following steps:

1. View the WLAN defaults.
2. Change the network mode.
3. Configure the SSID.
4. Configure the channel.
5. Configure the security mode.
6. Configure the passphrase.

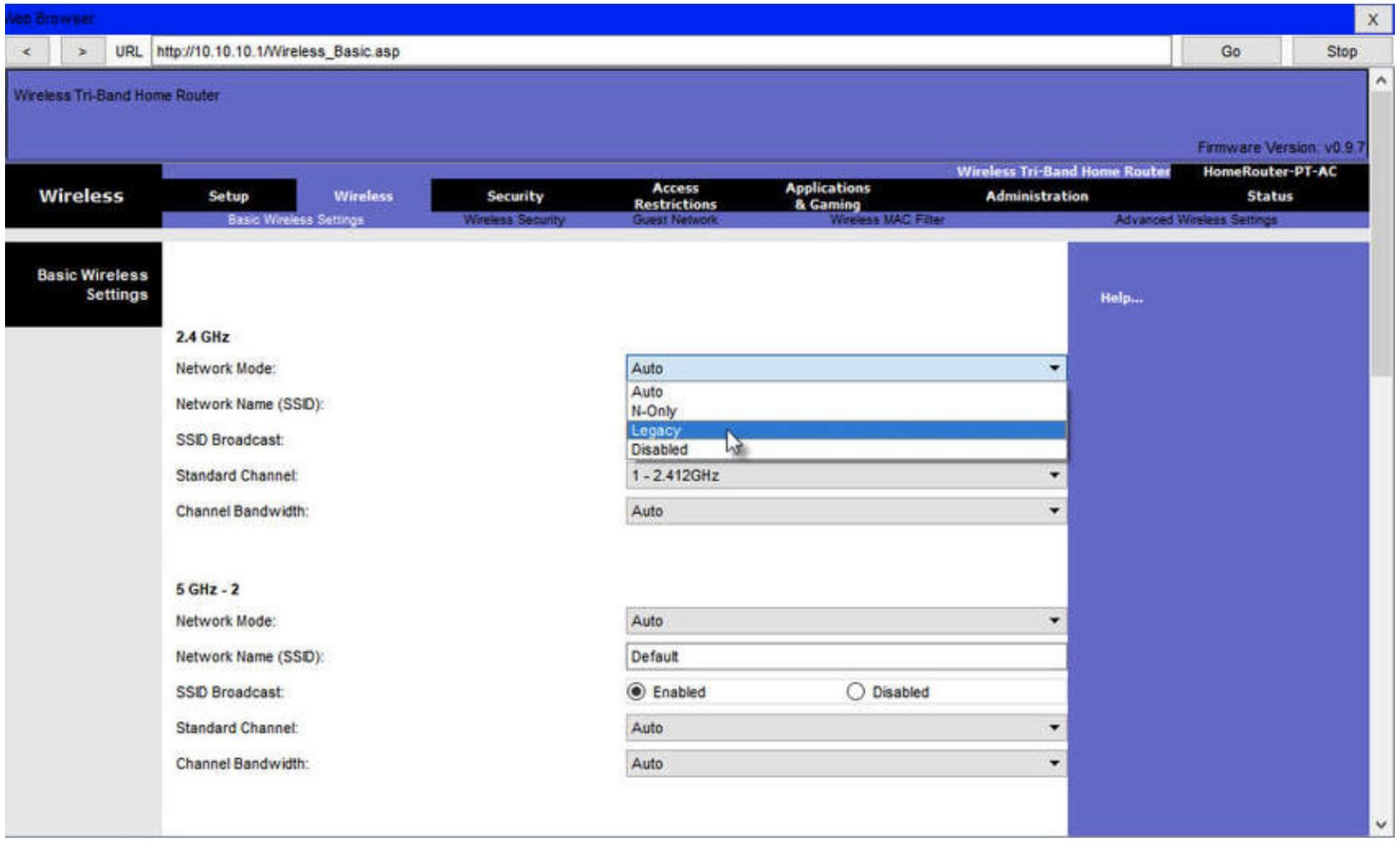
View the WLAN Defaults

- Out of the box, a wireless router provides wireless access to devices using a **default** wireless network **name** and **password**.
- The network name is called the **Service Set Identified (SSID)**.
- Locate the basic wireless settings for your router to change these defaults, as shown in the example.



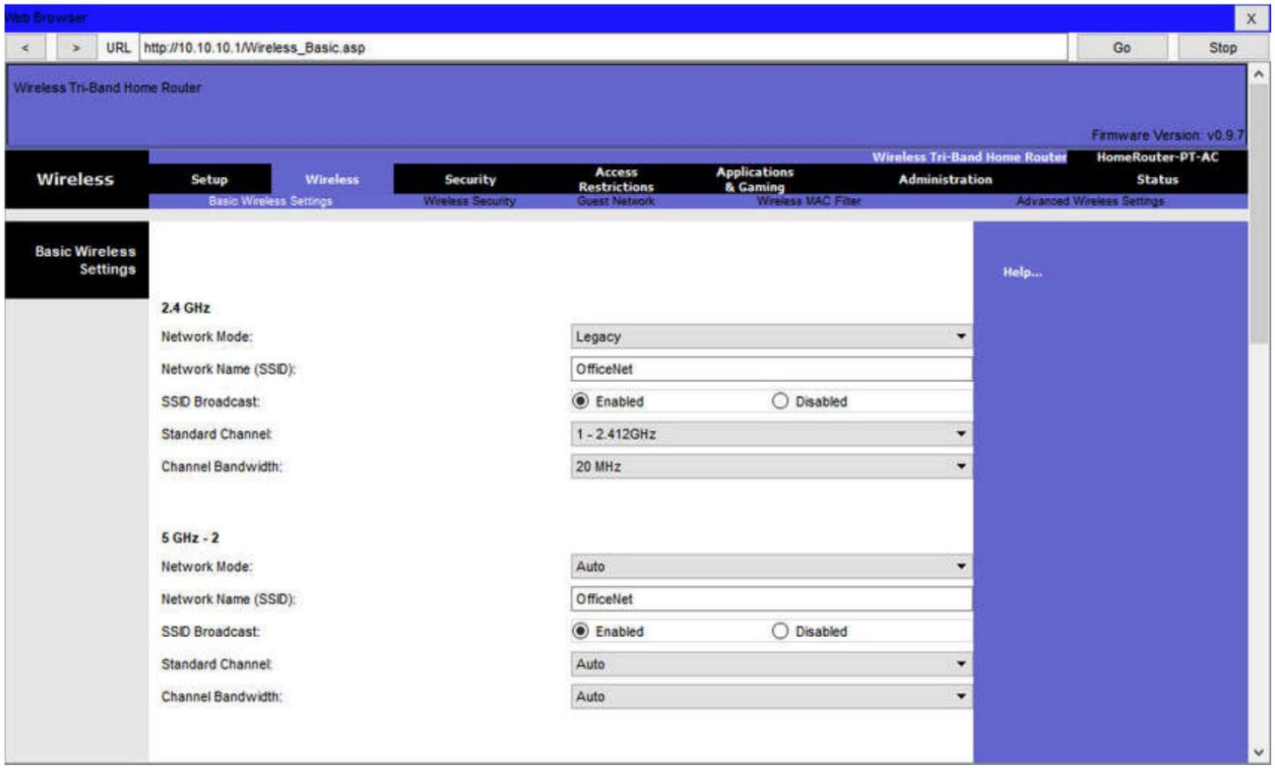
Change the Network Mode

- Some wireless routers allow you to select which 802.11 standard to implement. The example shows that "Legacy" has been selected.
- This means wireless devices connecting to the wireless router can have a variety of wireless NICs installed.
- Today's wireless routers configured for legacy or mixed mode most likely support 802.11a, 802.11n, and 802.11ac NICs.



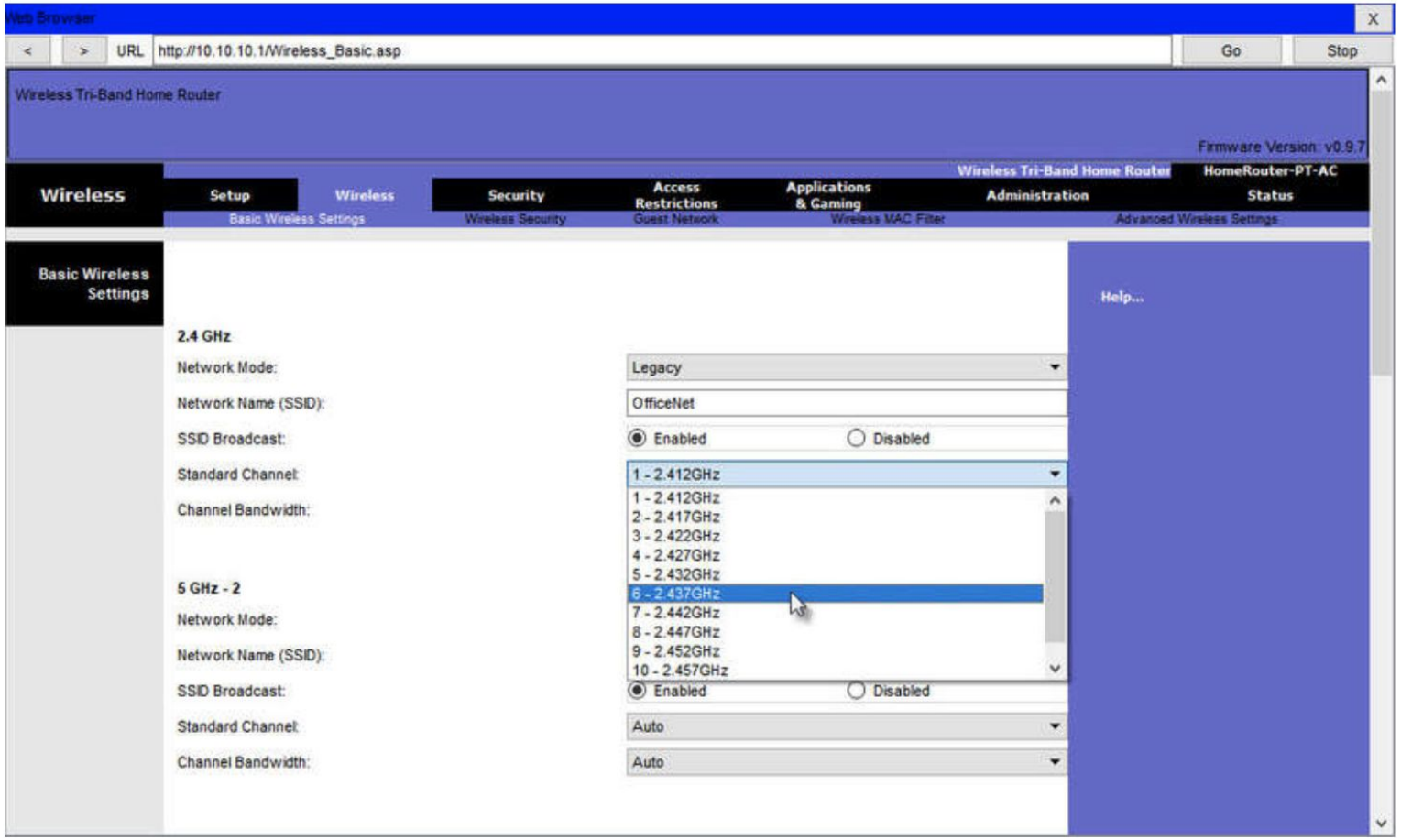
Configure the SSID

- Assign an SSID to the WLANs.
- The wireless router announces its presence by sending broadcasts advertising its SSID.
- This allows wireless hosts to automatically discover the name of the wireless network.
- If the SSID broadcast is disabled, you must manually enter the SSID on each wireless device that connects to the WLAN.



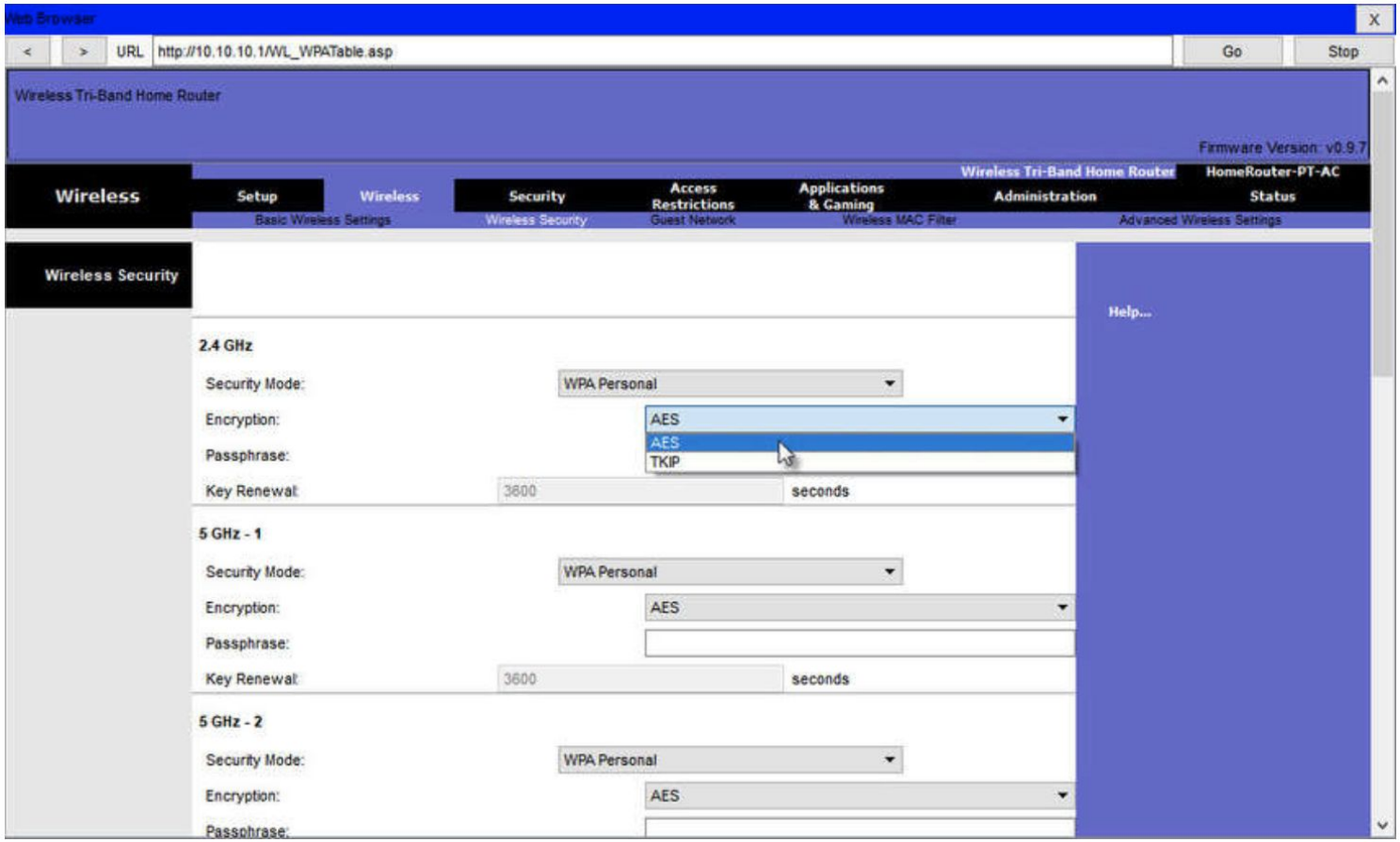
Configure the Channel

- To avoid interference is to configure non-overlapping channels on the wireless routers and access points that are near to each other.
- Specifically, channels 1, 6, and 11 are non-overlapping.
- In the example, the wireless router is configured to use channel 6.



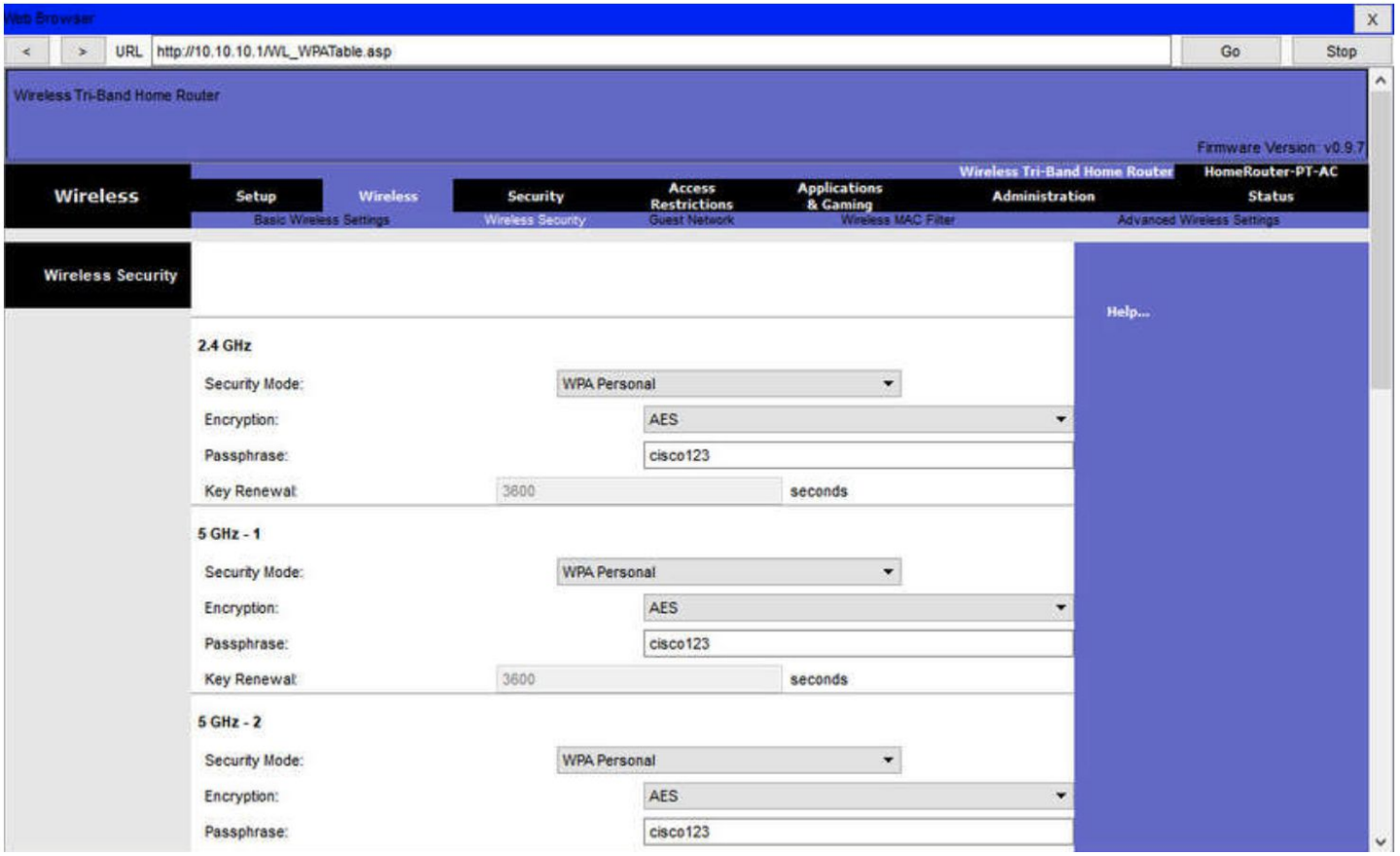
Configure the Security Mode

- Out of the box, a wireless router may have no WLAN security configured.
- In the example, the personal version of Wi-Fi Protected Access version 2 (WPA2 Personal) is selected for all three WLANs.
- WPA2 with Advanced Encryption Standard (AES) encryption is currently the strongest security mode.



Configure the Passphrase

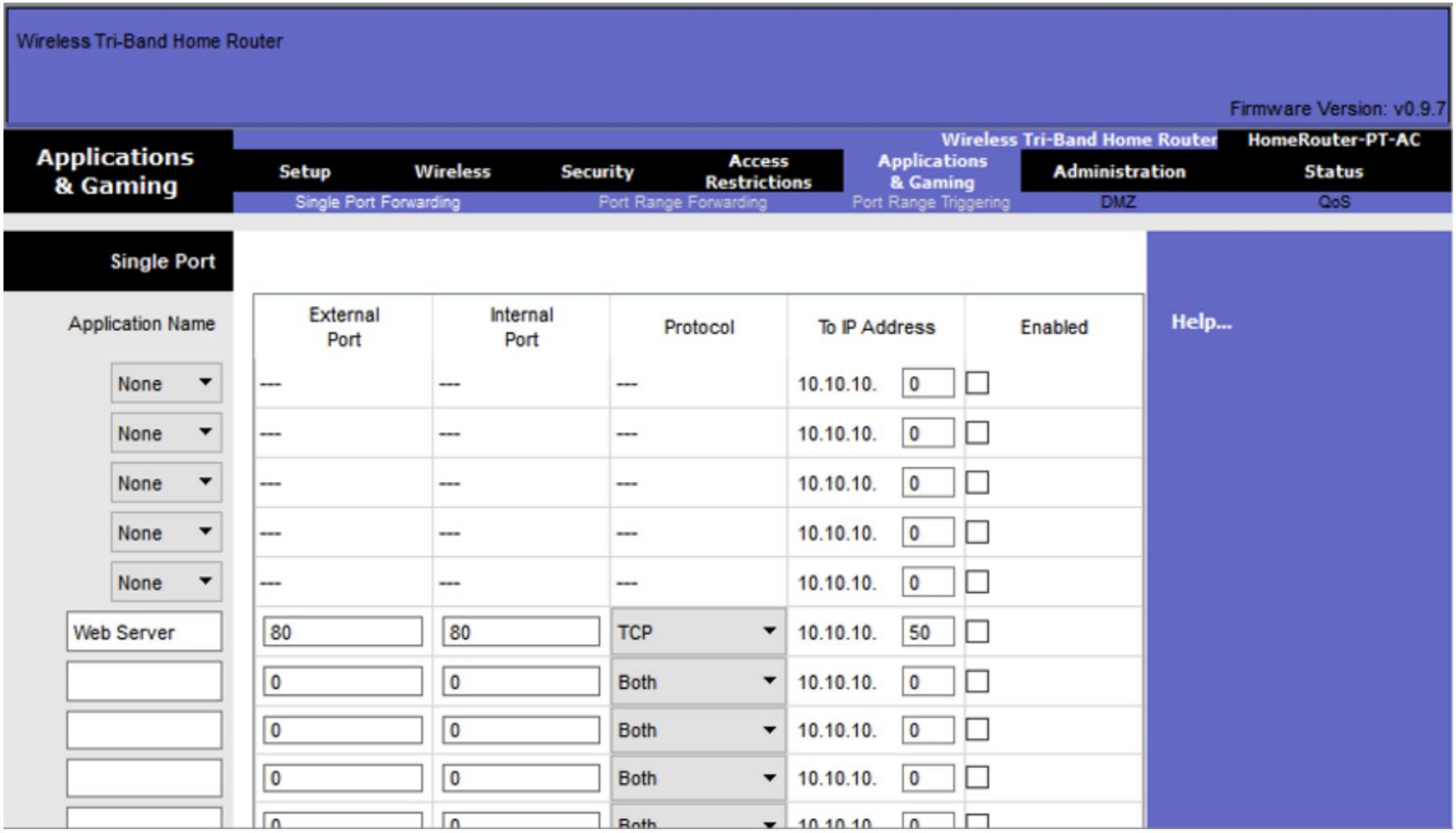
- WPA2 personal uses a passphrase to authenticate wireless clients.
- WPA2 personal is easier to use in a small office or home environment because it does not require an authentication server.
- Larger organizations implement WPA2 enterprise and require wireless clients to authenticate with a username and password.



Port Forwarding

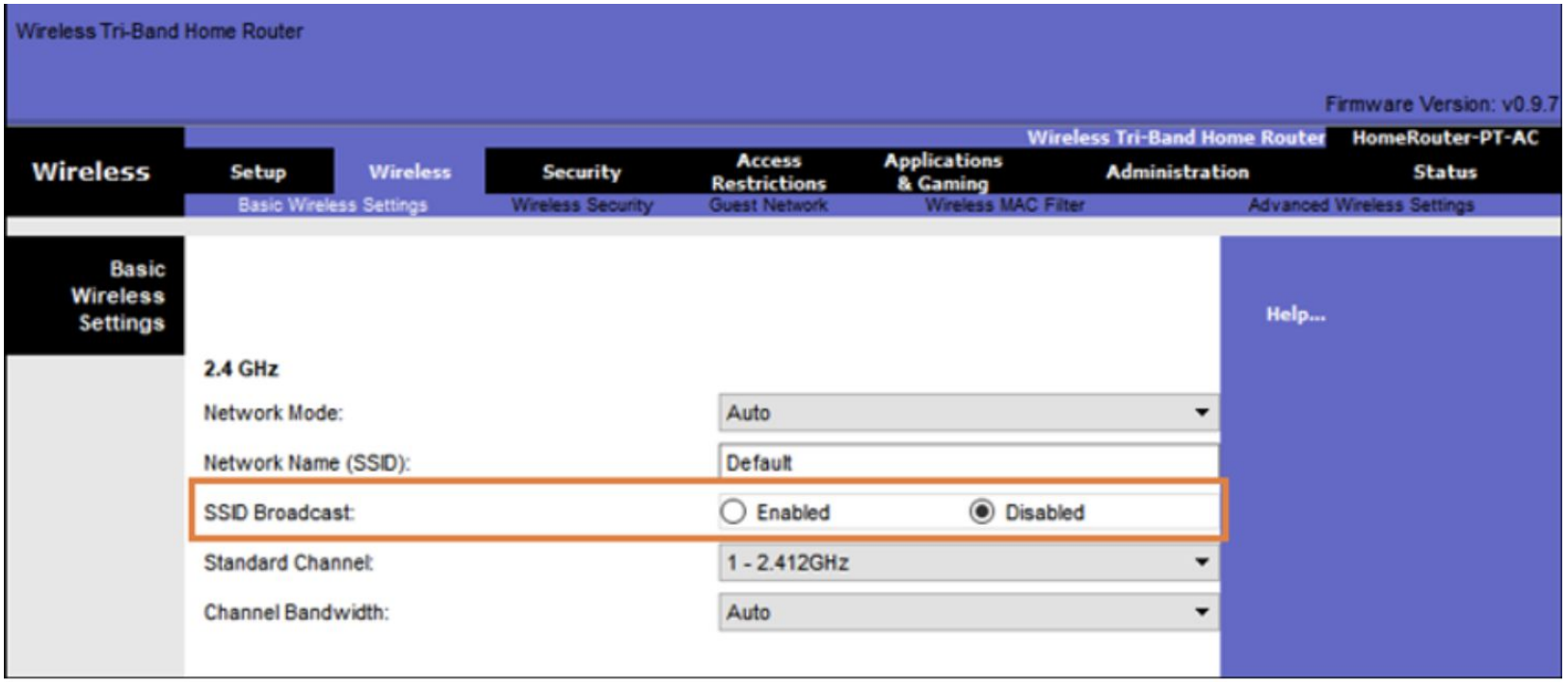
- Wireless routers typically block TCP and UDP ports to prevent unauthorized access in and out of a LAN.
- However, there are situations when specific ports must be opened so that certain programs and applications can communicate with devices on different networks.
- Port forwarding is a rule-based method of directing traffic between devices on separate networks.
- When traffic reaches the router, the router determines if the traffic should be forwarded to a certain device based on the port number found with the traffic. For example, a router might be configured to forward port 80, which is associated with HTTP.
- When the router receives a packet with the destination port of 80, the router forwards the traffic to the server inside the network that serves web pages.
- In the figure, port forwarding is enabled for port 80 and is associated with the web server at IPv4 address 10.10.10.50.

Port Forwarding (cont.)



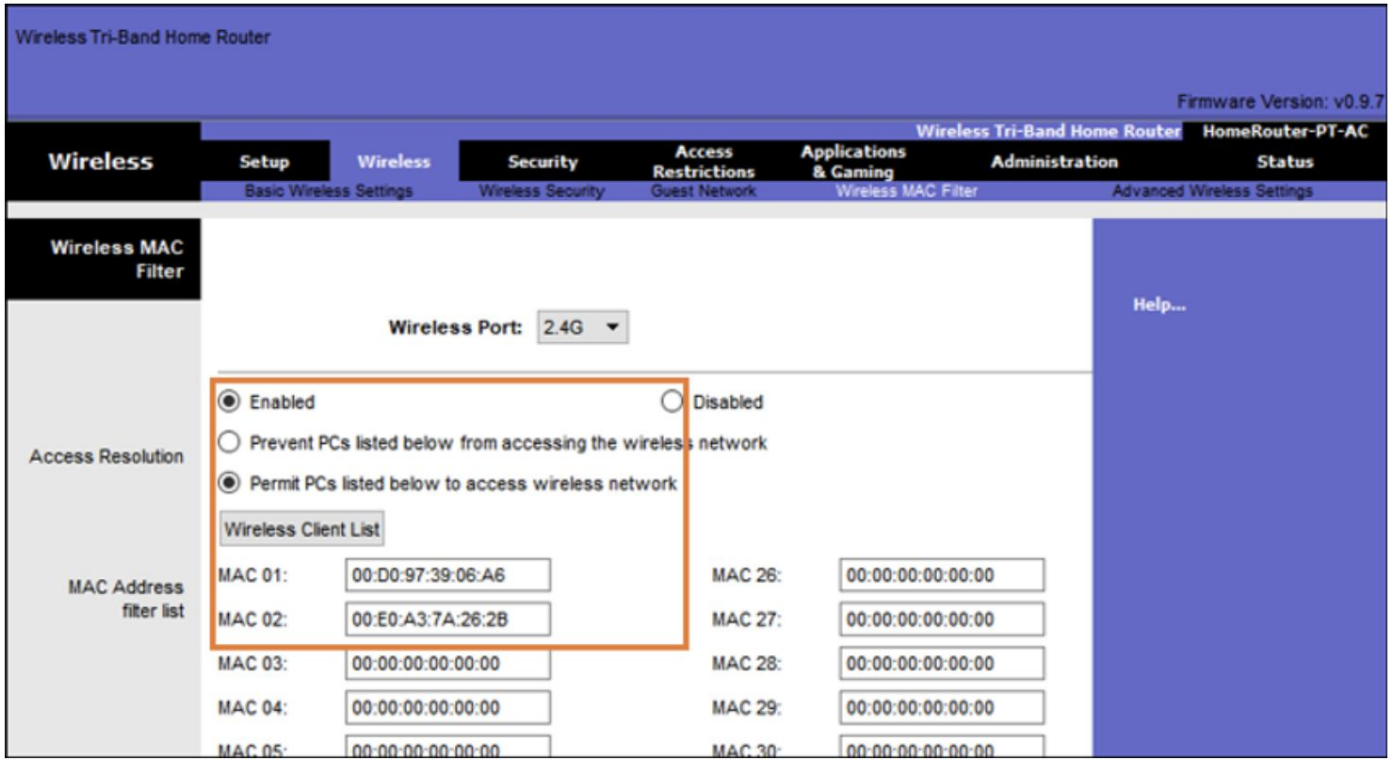
SSID Hiding

- A security feature
- APs and some wireless routers allow the SSID beacon frame to be disabled.
- Wireless clients must manually configure the SSID to connect to the network.



MAC Address Filtering

- An administrator can manually permit or deny clients wireless access based on their physical MAC hardware address.
- The router is configured to permit two MAC addresses.
- Devices with different MAC addresses will not be able to join the 2.4GHz WLAN.



COURSE: **NETWORK ADMINISTRATION AND MANAGEMENT**

COURSE INSTRUCTOR: **MIRALDA CUKA, PHD**

Lecture 9
Access Lists (ACLs)

What is ACL?

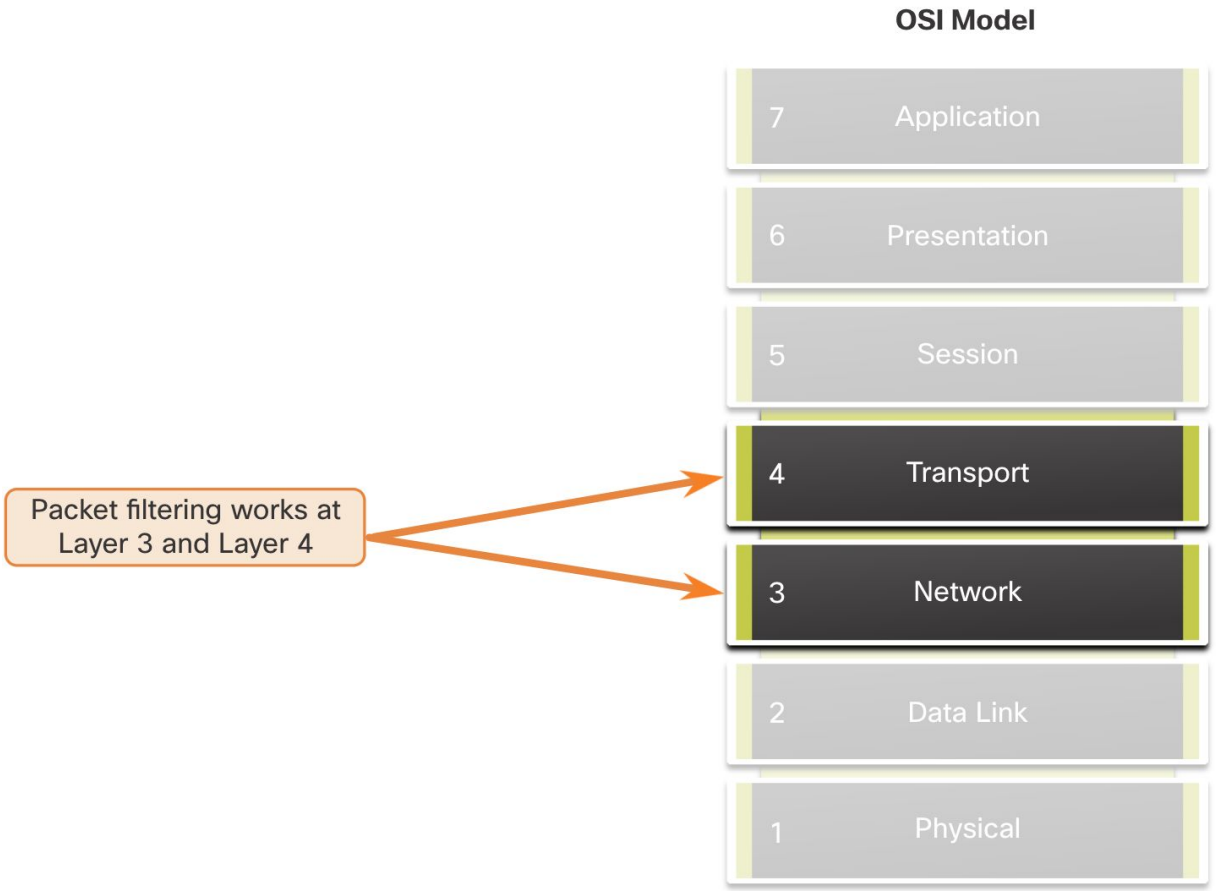
- Routers make routing decisions based on information in the packet header.
- Traffic entering a router interface is routed solely based on information within the routing table.
- The router compares the destination IP address with routes in the routing table to find the best match and then forwards the packet based on the best match route.
- That same process can be used to filter traffic using an access control list (ACL).
- When an ACL is applied to an interface, the router performs the additional task of evaluating all network packets as they pass through the interface to determine if the packet can be forwarded.

Why is ACL Used?

Task	Example
Limit network traffic to increase network performance	• A corporate policy prohibits video traffic on the network to reduce the network load. • A policy can be enforced using ACLs to block video traffic.
Provide traffic flow control	• A corporate policy requires that routing protocol traffic be limited to certain links only. • A policy can be implemented using ACLs to restrict the delivery of routing updates to only those that come from a known source.
Provide a basic level of security for network access	• Corporate policy demands that access to the Human Resources network be restricted to authorized users only. • A policy can be enforced using ACLs to limit access to specified networks.
Filter traffic based on traffic type	• Corporate policy requires that email traffic be permitted into a network, but that Telnet access be denied. • A policy can be implemented using ACLs to filter traffic by type.
Screen hosts to permit or deny access to network services	• Corporate policy requires that access to some file types (e.g., FTP or HTTP) be limited to user groups. • A policy can be implemented using ACLs to filter user access to services.
Provide priority to certain classes of network traffic	• Corporate traffic specifies that voice traffic be forwarded as fast as possible to avoid any interruption. • A policy can be implemented using ACLs and QoS services to identify voice traffic and process it immediately.

Packet Filtering

- Packet filtering controls access to a network by analyzing the incoming and/or outgoing packets and forwarding them or discarding them based on given criteria.
- Packet filtering can occur at Layer 3 or Layer 4, as shown in the figure.



Types of ACL

- Routers support two types of ACLs:
- Standard ACLs - ACLs only filter at Layer 3 using the source IPv4 address only.
- Extended ACLs - ACLs filter at Layer 3 using the source and / or destination IPv4 address. They can also filter at Layer 4 using TCP, UDP ports, and optional protocol type information for finer control.

ACL Operation

- ACLs define the set of rules that give added control for packets that enter inbound interfaces, packets that relay through the router, and packets that exit outbound interfaces of the router.
- ACLs can be configured to apply to inbound traffic and outbound traffic, as shown in the figure.

- Inbound ACL:
  - Filters packets before they are routed to the outbound interface.
  - Saves the overhead of routing lookups if the packet is discarded.
  - Best used to filter packets when the network attached to an inbound interface is the only source of packets that need to be examined.
- Outbound ACL:
  - Filters packets after being routed. Incoming packets are routed to the Best used when the same filter will be applied to packets coming from multiple inbound interfaces before exiting the same outbound interface.

Inbound ACL Example

- When an ACL is applied to an interface, it follows a specific operating procedure.
- The router extracts the source IPv4 address from the packet header.
- The router starts at the top of the ACL and compares the source IPv4 address to each ACE (access control entries) in a sequential order.
- When a match is made, the router carries out the instruction, either permitting or denying the packet.
- If the source IPv4 address does not match any ACEs in the ACL, the packet is discarded because there is an implicit deny ACE automatically applied to all ACLs.

Wildcard Mask Types

Wildcard to Match a Host

- The wildcard mask is used to match a specific host IPv4 address. Assume ACL 10 needs an ACE that only permits the host with IPv4 address 192.168.1.1.
- "0" equals a match and "1" equals ignore.
- To match a specific host IPv4 address, a wildcard mask consisting of all zeroes (i.e., 0.0.0.0) is required.
- The table lists in binary, the host IPv4 address, the wildcard mask, and the permitted IPv4 address.
- The 0.0.0.0 wildcard mask stipulates that every bit must match exactly. Therefore, when the ACE is processed, the wildcard mask will permit only the 192.168.1.1 address.
- The resulting ACE in ACL 10 would be access-list 10 permit 192.168.1.1 0.0.0.0.

	Decimal	Binary
IPv4 address	192.168.1.1	11000000.10101000.00000001.00000001
Wildcard Mask	0.0.0.0	00000000.00000000.00000000.00000000
Permitted IPv4 Address	**192.168.1.1**	**11000000.10101000.00000001.00000001**

Wildcard Mask Types

Wildcard Mask to Match an IPv4 Subnet

- In this example, ACL 10 needs an ACE that permits all hosts in the 192.168.1.0/24 network.
- The wildcard mask 0.0.0.255 stipulates that the very first three octets must match exactly but the fourth octet does not.
- The table lists in binary, the host IPv4 address, the wildcard mask, and the permitted IPv4 addresses.
- When processed, the wildcard mask 0.0.0.255 permits all hosts in the 192.168.1.0/24 network.
- The resulting ACE in ACL 10 would be access-list 10 permit 192.168.1.0 0.0.0.255.

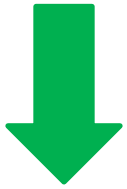
	Decimal	Binary
IPv4 address	192.168.1.1	11000000.10101000.00000001.00000001
Wildcard Mask	0.0.0.255	00000000.00000000.00000000.11111111
Permitted Host IPv4 Addresses	**192.168.1.1** to **192.168.1.254**	**11000000.10101000.00000001.00000000** to **11000000.10101000.00000001.11111111**

Wildcard Mask Keywords

The two keywords are:

- **host** - This keyword substitutes for the 0.0.0.0 mask. This mask states that all IPv4 address bits must match to filter just one host address.
- **any** - This keyword substitutes for the 255.255.255.255 mask. This mask says to ignore the entire IPv4 address or to accept any addresses.

```
R1(config)# access-list 10 permit 192.168.10.10  0.0.0.0
R1(config)# access-list 11 permit  0.0.0.0 255.255.255.255
R1(config)#
```



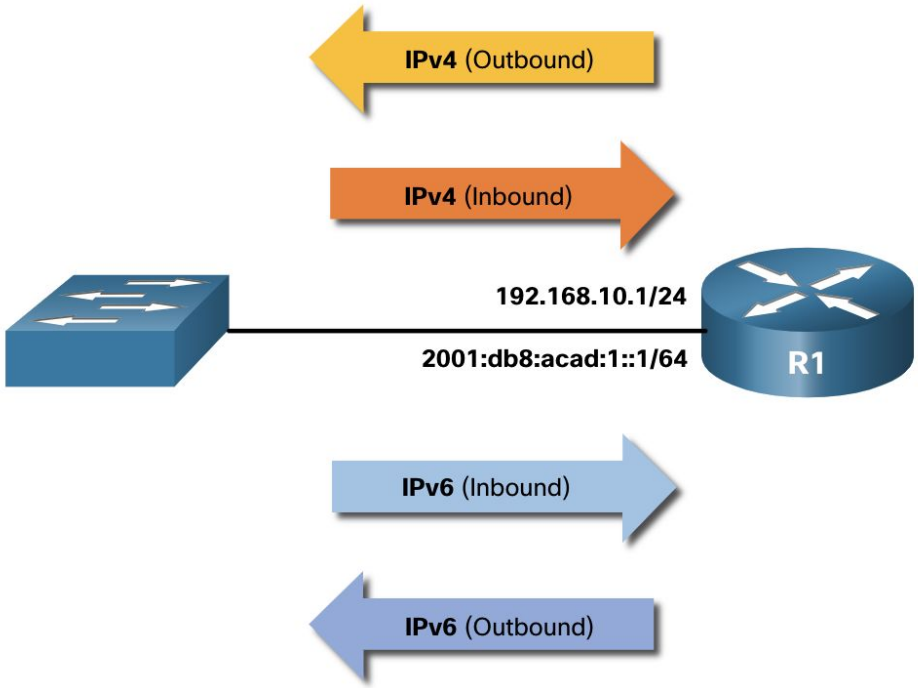
```
R1(config)# access-list 10 permit  host  192.168.10.10
R1(config)# access-list 11 permit  any
R1(config)#
```

ACL Interfaces

- There is a limit on the number of ACLs that can be applied on a router interface.

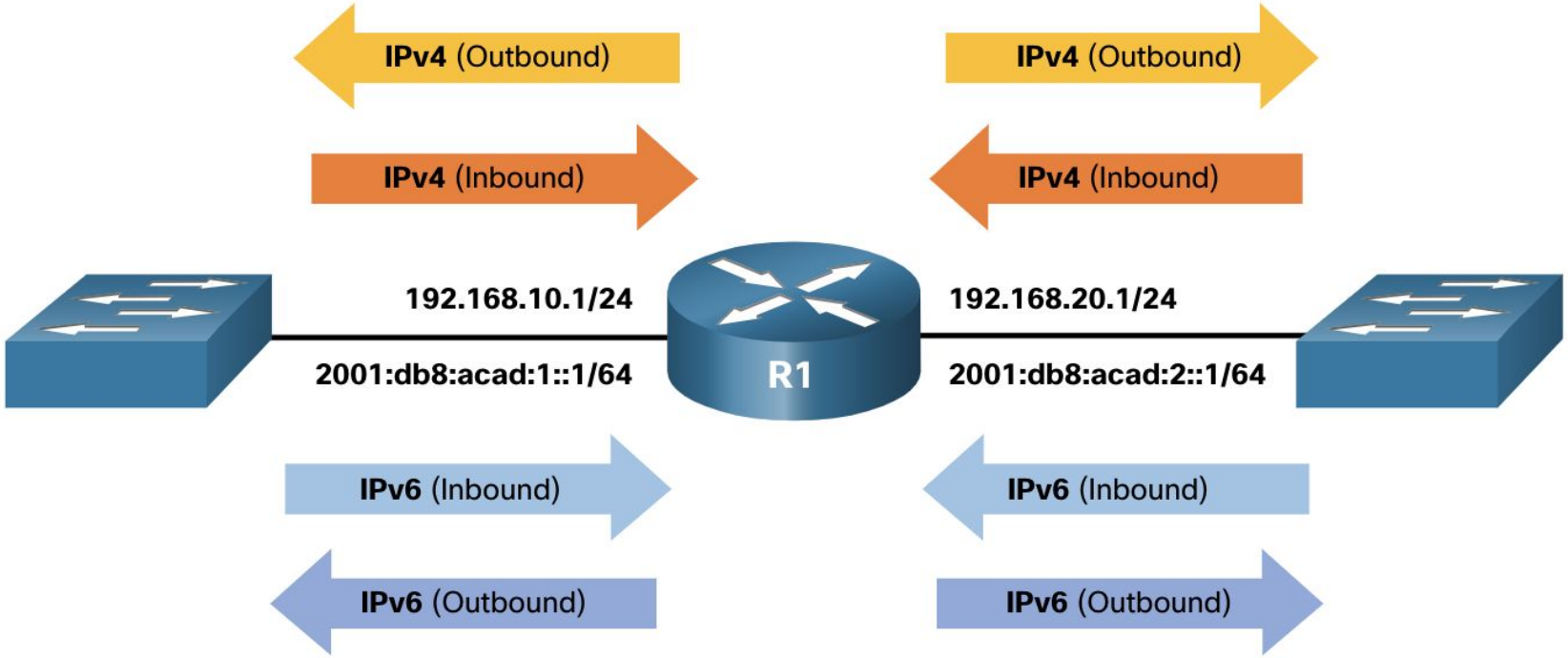
  Specifically, a router interface can have:

  - one outbound IPv4 ACL
  - one inbound IPv4 ACL
  - one inbound IPv6 ACL
  - one outbound IPv6 ACL



ACLs per Interface

- A router could have up to 8 ACLs configured and applied to interfaces.
- Each interface would have four ACLs; two ACLs for IPv4 and two ACLs for IPv6.
- For each protocol, one ACL is for inbound traffic and one for outbound traffic.



Note: ACLs do not have to be configured in both directions. The number of ACLs and their direction applied to the interface will depend on the security policy of the organization.

Write ACL before applying them to a router.

Types of ACL

- There are two types of IPv4 ACLs:
  - Standard ACLs - These permit or deny packets based only on the source IPv4 address.
  - Extended ACLs - These permit or deny packets based on the source IPv4 address and destination IPv4 address, protocol type, source and destination TCP or UDP ports and more.

```
R1(config)# access-list 10 permit 192.168.10.0 0.0.0.255
R1(config)#
```

- ACL 10 permits hosts on the source network 192.168.10.0/24.
- Because of the implied "deny any" at the end, all traffic except for traffic coming from the 192.168.10.0/24 network is blocked with this ACL.

```
R1(config)# access-list 100 permit tcp 192.168.10.0 0.0.0.255 any eq www
R1(config)#
```

- An extended ACL 100 permits traffic originating from any host on the 192.168.10.0/24 network to any IPv4 network if the destination host port is 80 (HTTP).

Numbered ACLs

- ACLs number 1 to 99, or 1300 to 1999 are standard ACLs while ACLs number 100 to 199, or 2000 to 2699 are extended ACLs.

```
R1(config)# access-list ?
  <1-99>        IP standard access list
  <100-199>     IP extended access list
  <1100-1199>   Extended 48-bit MAC address access list
  <1300-1999>   IP standard access list (expanded range)
  <200-299>     Protocol type-code access list
  <2000-2699>   IP extended access list (expanded range)
  <700-799>     48-bit MAC address access list
  rate-limit    Simple rate-limit specific access list
  template      Enable IP template acls
R1(config)# access-list
```

Named ACLs

- Named ACLs is the preferred method to use when configuring ACLs.
- Specifically, standard and extended ACLs can be named to provide information about the purpose of the ACL.
- For example, naming an extended ACL FTP-FILTER is far better than having a numbered ACL 100.
- The ip access-list global configuration command is used to create a named ACL.

```
R1(config)# ip access-list extended FTP-FILTER
R1(config-ext-nacl)# permit tcp 192.168.10.0 0.0.0.255 any eq ftp
R1(config-ext-nacl)# permit tcp 192.168.10.0 0.0.0.255 any eq ftp-data
R1(config-ext-nacl)#
```

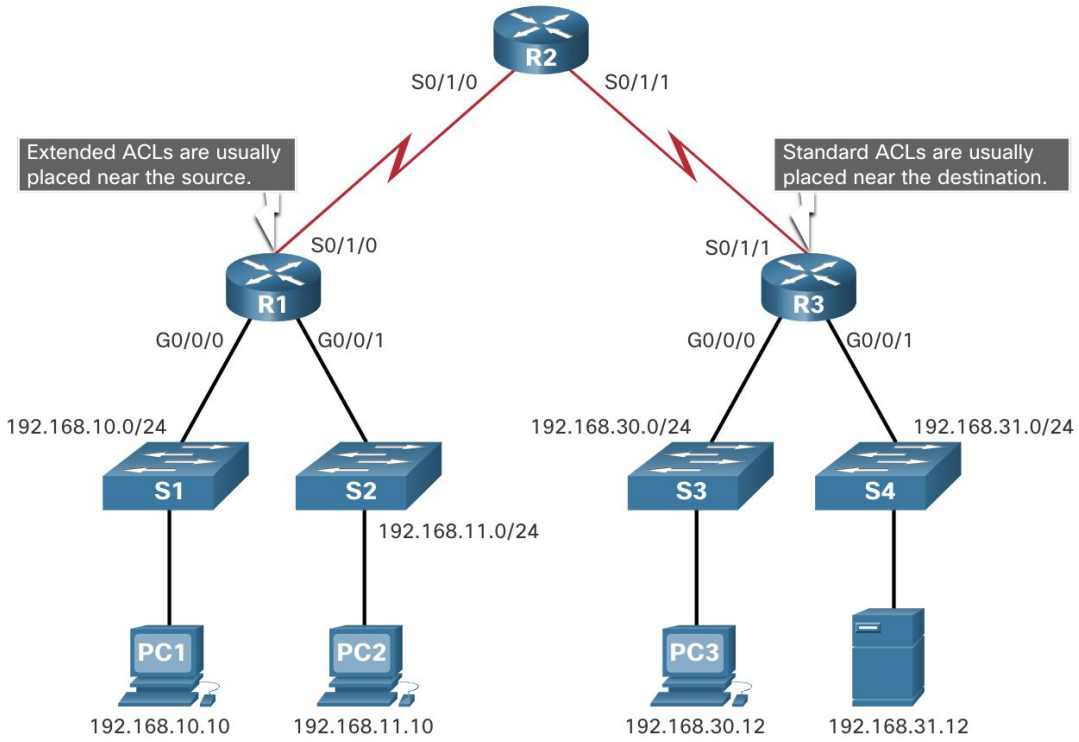
Naming ACLs

Rules to follow for naming ACLs:

- Assign a name to identify the purpose of the ACL.
- Names can contain alphanumeric characters.
- Names cannot contain spaces or punctuation.
- It is suggested that the name be written in CAPITAL LETTERS.
- Entries can be added or deleted within the ACL.

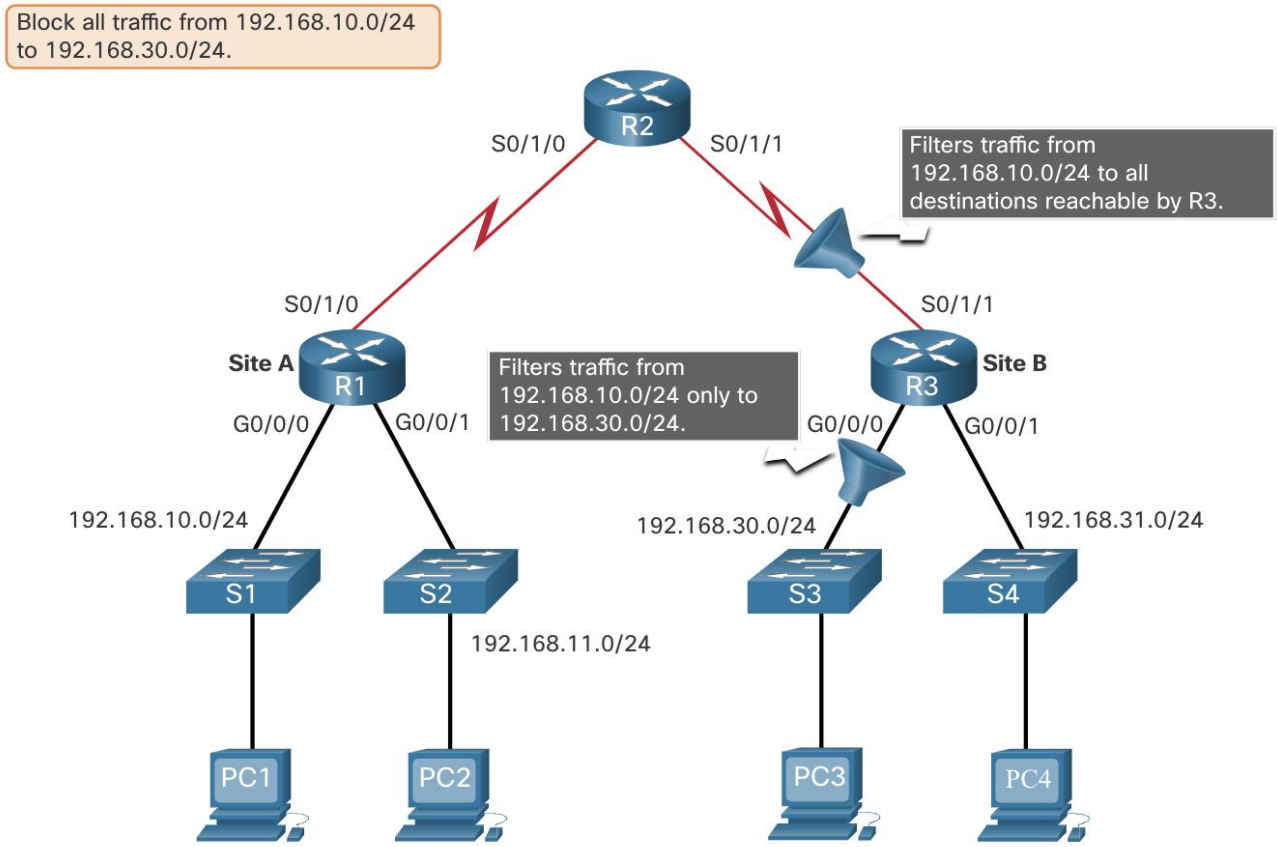
Where to Place ACLs?

- Every ACL should be placed where it has the greatest impact on efficiency.
- The figure illustrates where standard and extended ACLs should be located in an enterprise network.
- Assume the objective to prevent traffic originating in the 192.168.10.0/24 network from reaching the 192.168.30.0/24 network.



Standard ACL Placement Example

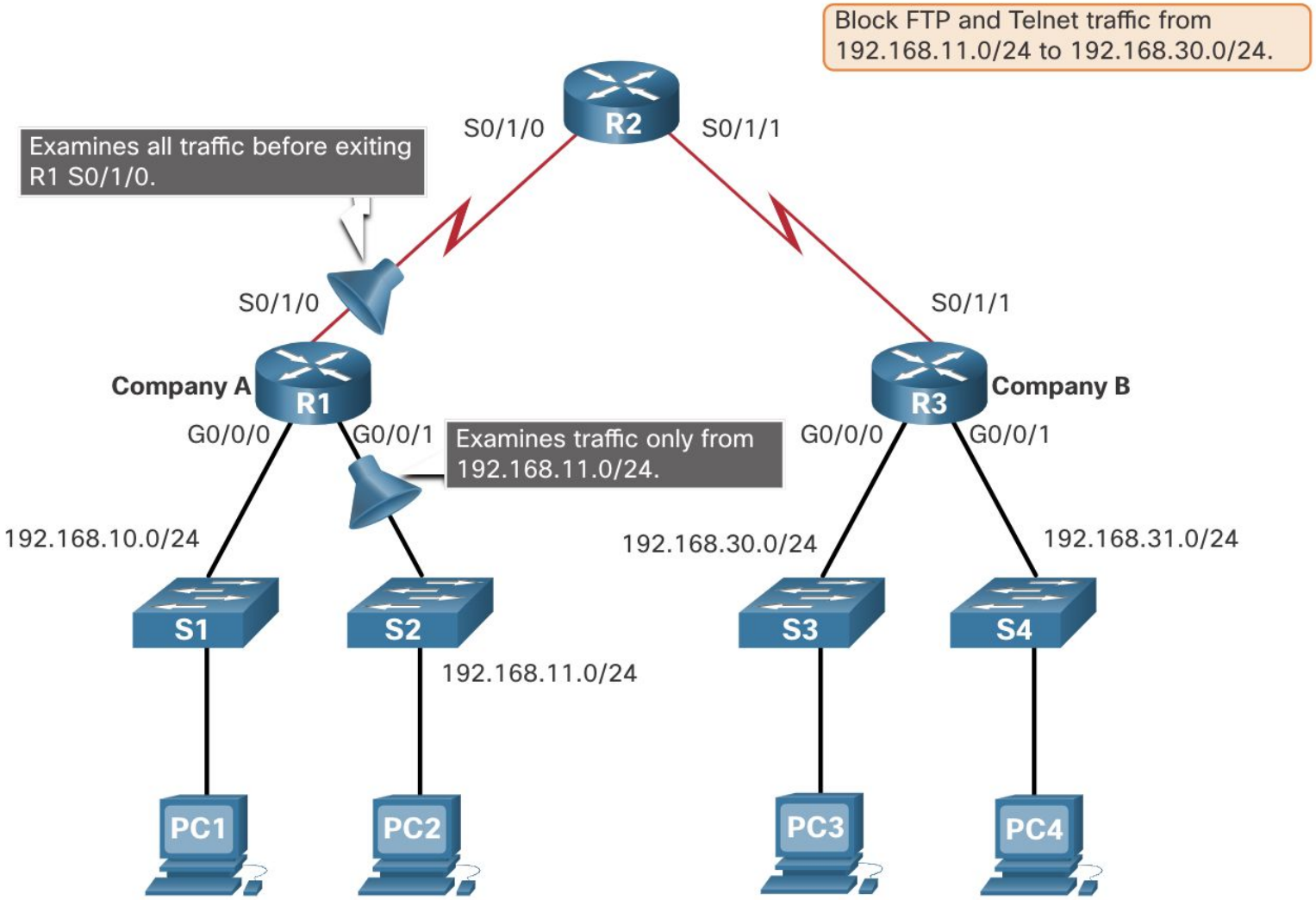
- Standard ACLs should be located as close to the destination as possible.
- In the figure, the administrator wants to prevent traffic originating in the 192.168.10.0/24 network from reaching the 192.168.30.0/24 network.



Extended ACL Placement Example

- Extended ACL should be located as close to the source as possible.
- This prevents unwanted traffic from being sent across multiple networks only to be denied when it reaches its destination.
- However, the organization can only place ACLs on devices that they control. Therefore, the extended ACL placement must be determined in the context of where organizational control extends.
- In the figure, for example, Company A wants to deny Telnet and FTP traffic to Company B's 192.168.30.0/24 network from their 192.168.11.0/24 network while permitting all other traffic.

Extended ACL Placement Example (cont.)



Configuring Numbered Standard ACL

- To create a numbered standard ACL, use the following global configuration command:

```
Router(config)# access-list access-list-number {deny | permit | remark text} source [source-wildcard] [log]
```

- Use the no access-list *access-list-number* global configuration command to remove a numbered standard ACL.

Parameter	Description
access-list-number	• This is the decimal number of the ACL. • Standard ACL number range is 1 to 99 or 1300 to 1999.
deny	This denies access if the condition is matched.
permit	This permits access if the condition is matched.
remark *text*	• (Optional) This adds a text entry for documentation purposes. • Each remark is limited to 100 characters.
source	• This identifies the source network or host address to filter. • Use the **any** keyword to specify all networks. • Use the **host** *ip-address* keyword or simply enter an *ip-address* (without the **host** keyword) to identify a specific IP address.
source-wildcard	(Optional) This is a 32-bit wildcard mask that is applied to the *source*. If omitted, a default 0.0.0.0 mask is assumed.
log	• (Optional) This keyword generates and sends an informational message whenever the ACE is matched. • Message includes ACL number, matched condition (i.e., permitted or denied), source address, and number of packets. • This message is generated for the first matched packet. • This keyword should only be implemented for troubleshooting or security reasons.

Configuring Named Standard ACL

- Naming an ACL makes it easier to understand its function. To create a named standard ACL, use the following global configuration command:

```
Router(config)# ip access-list standard access-list-name
```

- Note: Use the no ip access-list standard *access-list-name* global configuration command to remove a named standard IPv4 ACL.

Applying Standard ACL to Interface

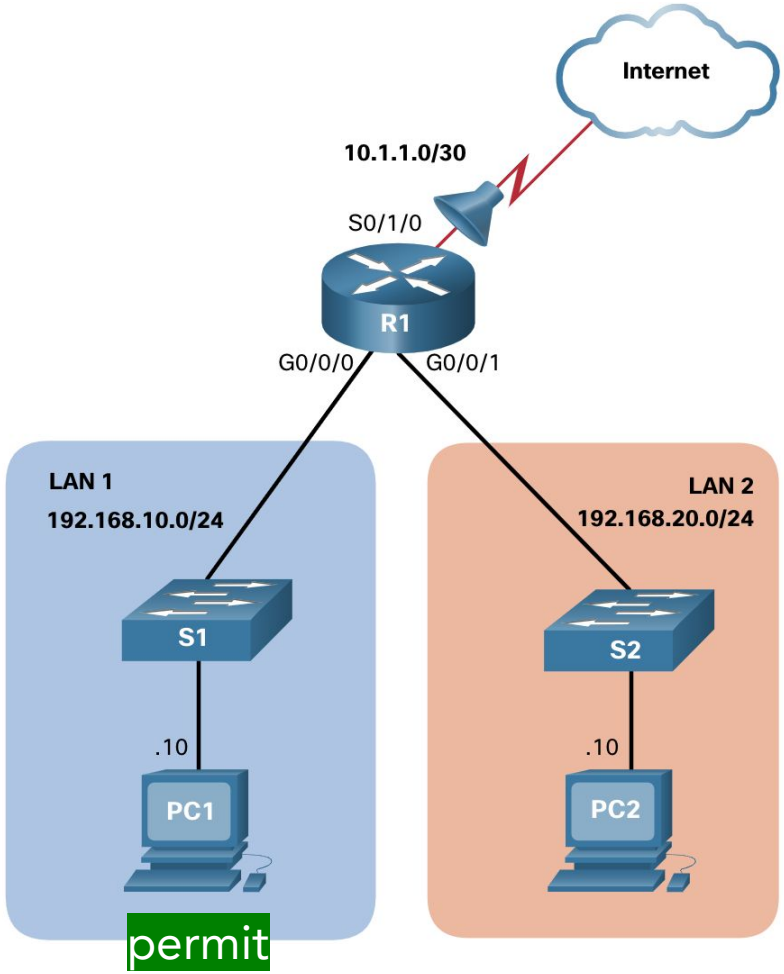
- After a standard IPv4 ACL is configured, it must be linked to an interface or feature.
- The following command can be used to bind a numbered or named standard IPv4 ACL to an interface:

```
Router(config-if) # ip access-group {access-list-number | access-list-name} {in | out}
```

- To remove an ACL from an interface, first enter the **no ip access-group** interface configuration command. However, the ACL will still be configured on the router. To remove the ACL from the router, use the **no access-list** global configuration command.

Numbered Standard IPv4 ACL Example

- Assume only PC1 is allowed out to the internet.
- To enable this policy, a standard ACL ACE could be applied outbound on S0/1/0, as shown in the figure.



Numbered Standard IPv4 ACL Example (cont.)

- Assume only PC1 is allowed out to the internet.
- To enable this policy, a standard ACL ACE could be applied outbound on S0/1/0, as shown in the figure.

```
R1(config)# access-list 10 remark ACE permits ONLY host 192.168.10.10 to the internet
R1(config)# access-list 10 permit host 192.168.10.10
R1(config)# do show access-lists
Standard IP access list 10
    10 permit 192.168.10.10
R1(config)#
```

- Remark command is not required to enable the ACL, it is strongly suggested for documentation purposes.
- Now assume that a new network policy states that hosts in LAN 2 should also be permitted to the internet.
- To enable this policy, a second standard ACL ACE could be added to ACL 10, as shown in the output.

```
R1(config)# access-list 10 remark ACE permits all host in LAN 2
R1(config)# access-list 10 permit 192.168.20.0 0.0.0.255
R1(config)# do show access-lists
Standard IP access list 10
    10 permit 192.168.10.10
    20 permit 192.168.20.0, wildcard bits 0.0.0.255
R1(config)#
```

Numbered Standard IPv4 ACL Example (cont.)

- Apply ACL 10 outbound on the Serial 0/1/0 interface.

```
R1(config)# interface Serial 0/1/0
R1(config-if)# ip access-group 10 out
R1(config-if)# end
R1#
```

Named Standard IPv4 ACL Example

- Assume only PC1 is allowed out to the internet.
- To enable this policy, a named standard ACL called PERMIT-ACCESS could be applied outbound on S0/1/0.
- Remove the previously configured named ACL 10 and create a named standard ACL called PERMIT-ACCESS, as shown here:

```
R1(config)# no access-list 10
R1(config)# ip access-list standard PERMIT-ACCESS
R1(config-std-nacl)# remark ACE permits host 192.168.10.10
R1(config-std-nacl)# permit host 192.168.10.10
R1(config-std-nacl)#
```

- Now add an ACE permitting only host 192.168.10.10 and another ACE permitting all LAN 2 hosts to the internet:

```
R1(config-std-nacl)# remark ACE permits host 192.168.10.10
R1(config-std-nacl)# permit host 192.168.10.10
R1(config-std-nacl)# remark ACE permits all hosts in LAN 2
R1(config-std-nacl)# permit 192.168.20.0 0.0.0.255
R1(config-std-nacl)# exit
R1(config)#
```

Named Standard IPv4 ACL Example (cont.)

- Apply the new named ACL outbound to the Serial 0/1/0 interface.

```
R1(config)# interface Serial 0/1/0
R1(config-if)# ip access-group PERMIT-ACCESS out
R1(config-if)# end
R1#
```

Configure Extended ACL

- Extended ACLs are used more often than standard ACLs because they provide a greater degree of control.
- They can filter on source address, destination address, protocol (i.e., IP, TCP, UDP, ICMP), and port number.
- This provides a greater range of criteria on which to base the ACL.
- An extended ACL can allow email traffic from a network to a specific destination while denying file transfers and web browsing.
- Like standard ACLs, extended ACLs can be created as:
- Numbered Extended ACL - Created using the **access-list** *access-list-number* global configuration command.
- Named Extended ACL - Created using the **ip access-list extended** *access-list-name*.

Numbered ACL Syntax

- The procedural steps for configuring extended ACLs are the same as for standard ACLs. The extended ACL is first configured, and then it is activated on an interface.
- However, the command syntax and parameters are more complex to support the additional features provided by extended ACLs.
- To create a numbered extended ACL, use the following global configuration command:

```
Router(config)# access-list access-list-number {deny | permit | remark text} protocol source
source-wildcard [operator {port}] destination destination-wildcard [operator {port}]
[established] [log]
```

- Use the no access-list *access-list-number* global configuration command to remove an extended ACL.

Extended ACL Keywords and Parameters

- The table provides a detailed explanation of the syntax for an extended ACL.

Parameter	Description
access-list-number	• This is the decimal number of the ACL. • Extended ACL number range is 100 to 199 and 2000 to 2699.
deny	This denies access if the condition is matched.
permit	This permits access if the condition is matched.
remark *text*	• (Optional) Adds a text entry for documentation purposes. • Each remark is limited to 100 characters.
protocol	• Name or number of an internet protocol. • Common keywords include **ip**, **tcp**, **udp**, and **icmp**. • The **ip** keyword matches all IP protocols.
source	• This identifies the source network or host address to filter. • Use the **any** keyword to specify all networks. • Use the **host** *ip-address* keyword or simply enter an *ip-address* (without the **host** keyword) to identify a specific IP address.
source-wildcard	(Optional) A 32-bit wildcard mask that is applied to the source.
destination	• This identifies the destination network or host address to filter. • Use the **any** keyword to specify all networks. • Use the **host** *ip-address* keyword or *ip-address*.
destination-wildcard	(Optional) This is a 32-bit wildcard mask that is applied to the destination.
operator	• (Optional) This compares source or destination ports. • Some operators include **lt** (less than), **gt** (greater than), **eq** (equal), and **neq** (not equal).
port	(Optional) The decimal number or name of a TCP or UDP port.
established	• (Optional) For the TCP protocol only. • This is a 1st generation firewall feature.
log	• (Optional) This keyword generates and sends an informational message whenever the ACE is matched. • This message includes ACL number, matched condition (i.e., permitted or denied), source address, and number of packets. • This message is generated for the first matched packet. • This keyword should only be implemented for troubleshooting or security reasons.

Configure Numbered Extended ACL

- The command to apply an extended IPv4 ACL to an interface is the same as the command used for standard IPv4 ACLs.

```
Router(config-if)# ip access-group {access-list-number | access-list-name} {in | out}
```

- To remove an ACL from an interface, first enter the **no ip access-group** interface configuration command.
- To remove the ACL from the router, use the **no access-list** global configuration command.
- **Note**: The internal logic applied to the ordering of standard ACL statements does not apply to extended ACLs. The order in which the statements are entered during configuration is the order they are displayed and processed.

COURSE: **NETWORK ADMINISTRATION AND MANAGEMENT**

COURSE INSTRUCTOR: **MIRALDA CUKA, PHD**

Lecture 10
NAT&PAT

IPv4 Private Address Space

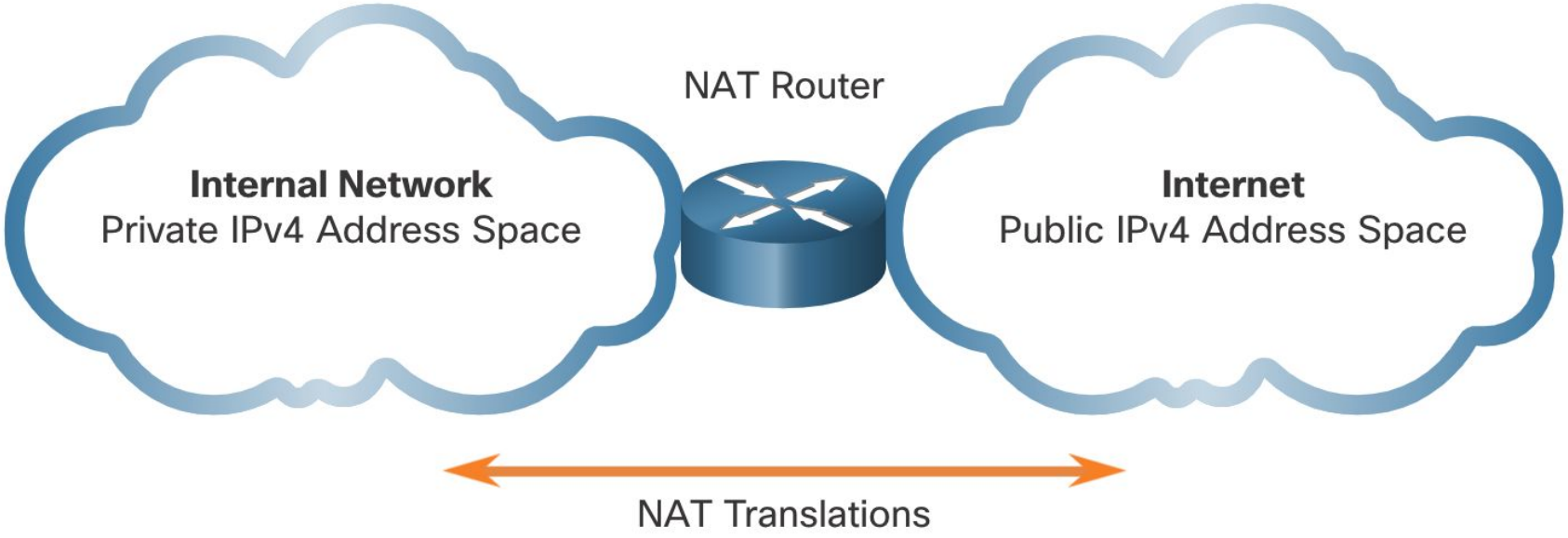
- Private IPv4 addresses are used within an organization or site to allow devices to communicate locally.
- Private IPv4 addresses cannot be routed over the internet.
- To allow a device with a private IPv4 address to access devices and resources outside of the local network, the private address must first be translated to a public address.
- NAT provides the translation of private addresses to public addresses.
- This allows a device with a private IPv4 address to access resources outside of their private network, such as those found on the internet. NAT, combined with private IPv4 addresses, has been the primary method of preserving public IPv4 addresses.
- A single, public IPv4 address can be shared by hundreds, even thousands of devices, each configured with a unique private IPv4 address.

IPv4 Private Address Space (cont.)

Class	RFC 1918 Internal Address Range	Prefix
A	10.0.0.0 - 10.255.255.255	10.0.0.0/8
B	172.16.0.0 - 172.31.255.255	172.16.0.0/12
C	192.168.0.0 - 192.168.255.255	192.168.0.0/16

Network Address Translator (NAT)

- Without NAT, the exhaustion of the IPv4 address space would have occurred already.
- NAT has limitations and disadvantages.
- The solution to the exhaustion of IPv4 address space and the limitations of NAT is the eventual transition to IPv6.



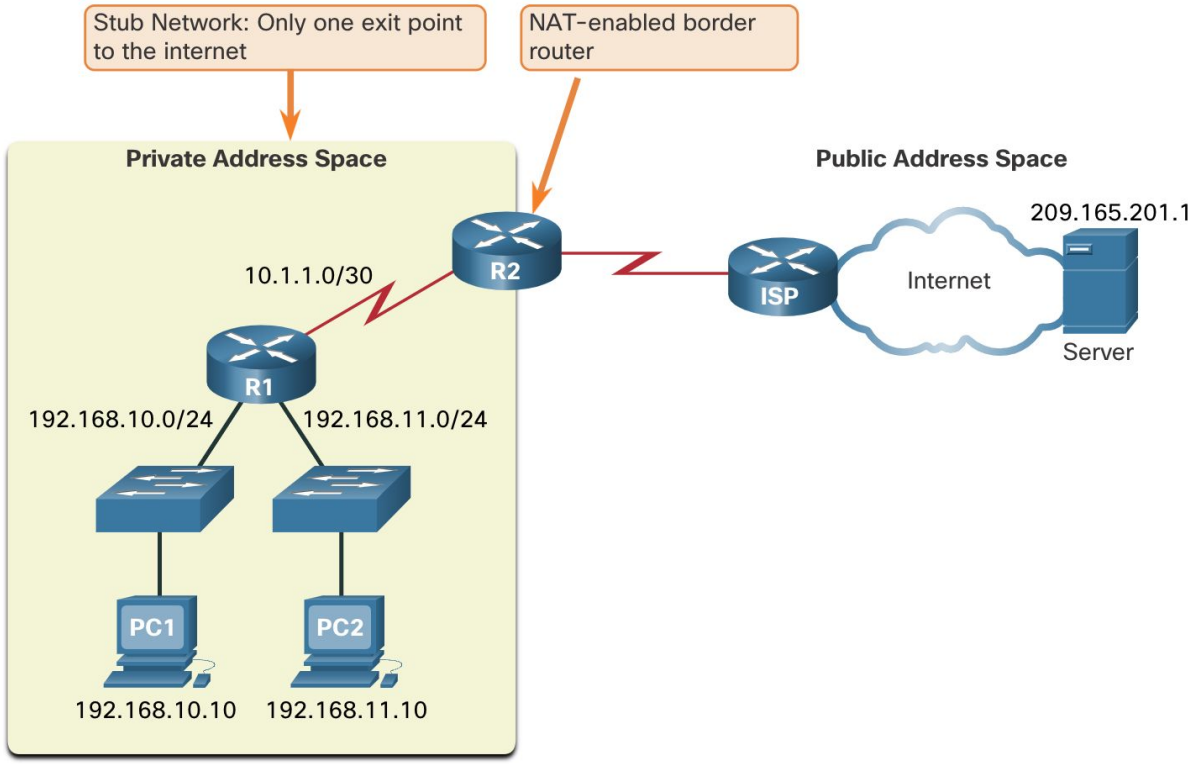
What is NAT?

- NAT has many uses, but its primary use is to conserve public IPv4 addresses.
- It allows networks to use *private IPv4 addresses internally* and providing translation to a *public address* only when needed.
- NAT adds a degree of privacy and security to a network, because it hides internal IPv4 addresses from outside networks.
- NAT-enabled routers can be configured with one or more valid public IPv4 addresses.
- These public addresses are known as the NAT pool.
- When an internal device sends traffic out of the network, the NAT-enabled router translates the internal IPv4 address of the device to a public address from the NAT pool.

What is NAT? (cont.)

- To outside devices, all traffic entering and exiting the network appears to have a public IPv4 address from the provided pool of addresses.
- A NAT router typically operates at the border of a stub network.

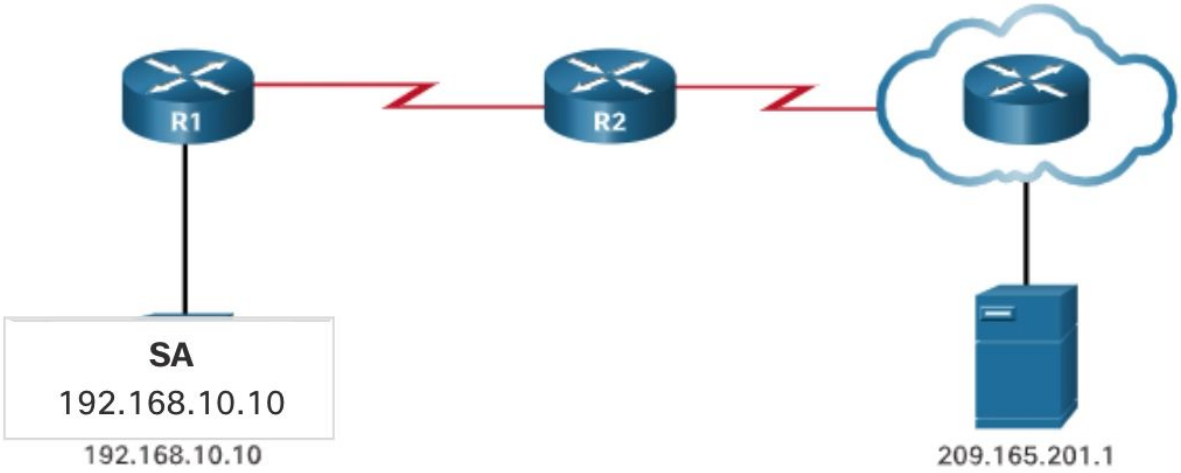
When a device inside the stub network wants to communicate with a device outside of its network, the packet is forwarded to the border router. The border router performs the NAT process.



Note: The connection to the ISP may use a private address or a public address that is shared among customers. For the purposes of this module, a public address is shown.

How NAT Works

- In the example below, PC1 with private address 192.168.10.10 wants to communicate with an outside web server with public address 209.165.201.1.



NAT Terminology

In NAT, terminology the inside network is the set of networks that is subject to translation.

The outside network refers to all other networks.

NAT includes four types of addresses:

- Inside local address
- Inside global address
- Outside local address
- Outside global address

NAT terminology is always applied from the perspective of the device with the translated address:

Inside address - The address of the device which is being translated by NAT.

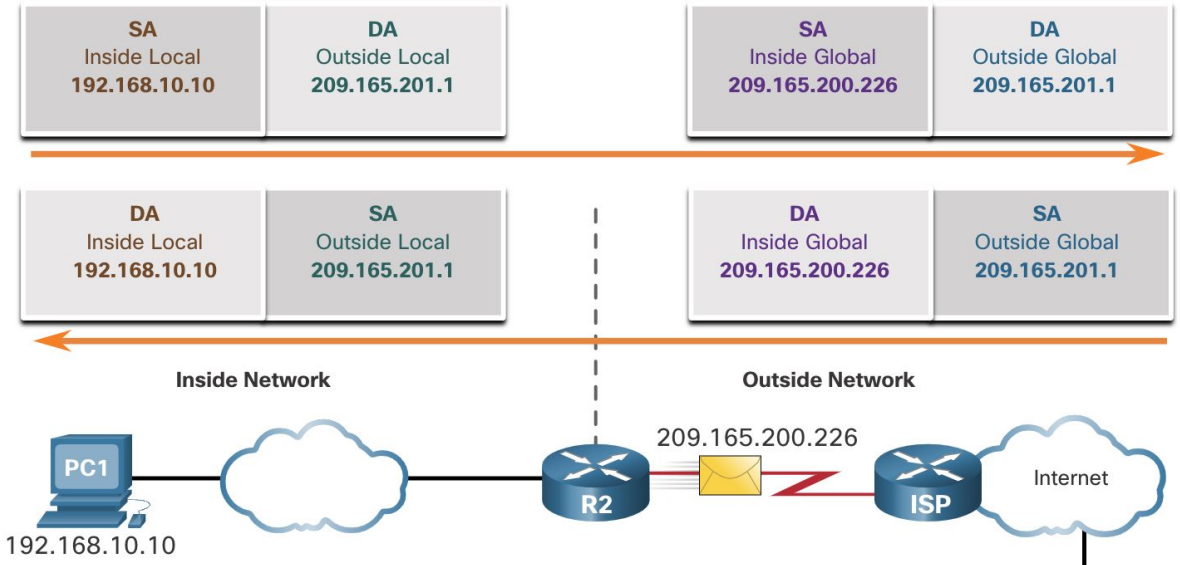
Outside address - The address of the destination device.

Local address - A local address is any address that appears on the inside portion of the network.

Global address - A global address is any address that appears on the outside portion of the network.

NAT Terminology (cont.)

- Inside and outside terms, are combined with the terms local and global to refer to specific addresses.
- The NAT router, R2 in the figure, is the demarcation point between the inside and outside networks.
- R2 is configured with a pool of public addresses to assign to inside hosts.



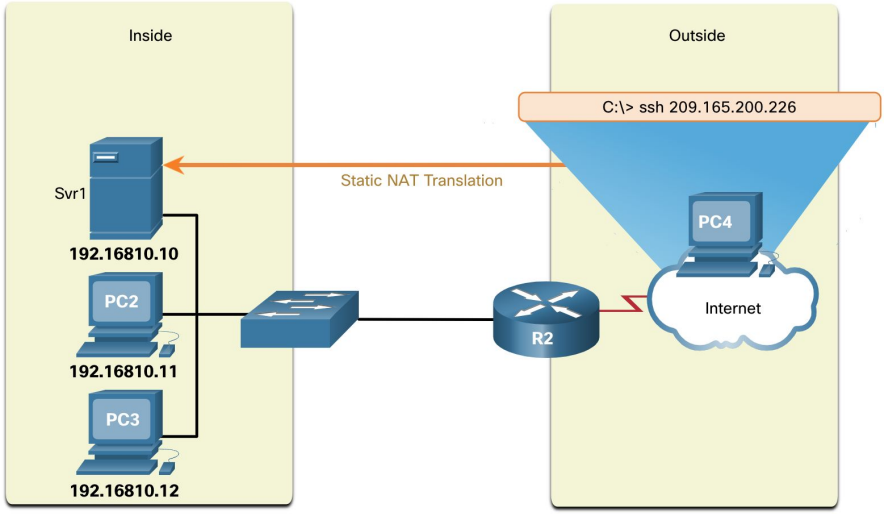
R2 NAT Table			
PC1		Web Server	
Inside Global Address	Inside Local Address	Outside Local Address	Outside Global Address
209.165.200.226	192.168.10.10	209.165.201.1	209.165.201.1

Types of NAT

- Static NAT
- Dynamic NAT
- PAT (Port Address Translator)

Static NAT

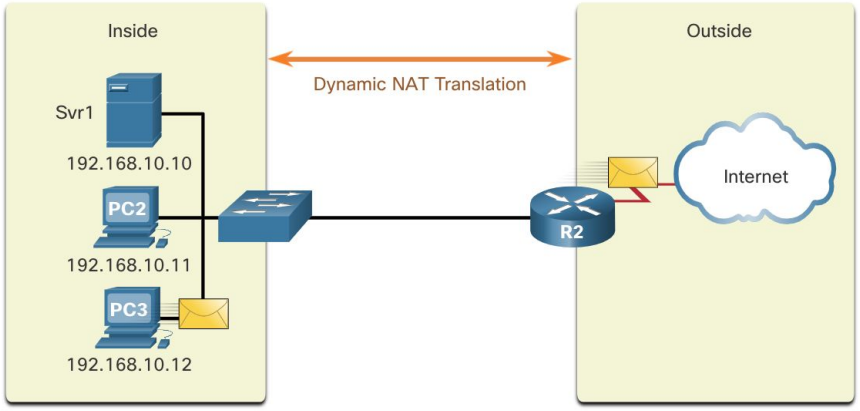
- Static NAT uses a one-to-one mapping of local and global addresses.
- These mappings are configured by the network administrator and remain constant.
- In the figure, R2 is configured with static mappings for the inside local addresses of Svr1, PC2, and PC3.
- When these devices send traffic to the internet, their **inside local addresses** are **translated** to the configured **inside global addresses**.
- To outside networks, these devices appear to have public IPv4 addresses.



Static NAT Table	
Inside Local Address	**Inside Global Address - Addresses reachable via R2**
192.168.10.10	209.165.200.226
192.168.10.11	209.165.200.227
192.168.10.12	209.165.200.228

Dynamic NAT

- Dynamic NAT uses a pool of public addresses and assigns them on a first-come, first-served basis.
- When an inside device requests access to an outside network, dynamic NAT assigns an available public IPv4 address from the pool.
- Similar to static NAT, dynamic NAT requires that enough public addresses are available to satisfy the total number of simultaneous user sessions.



IPv4 NAT Pool	
Inside Local Address	**Inside Global Address Pool - Addresses reachable via R2**
192.168.10.12	209.165.200.226
Available	209.165.200.227
Available	209.165.200.228
Available	209.165.200.229
Available	209.165.200.230

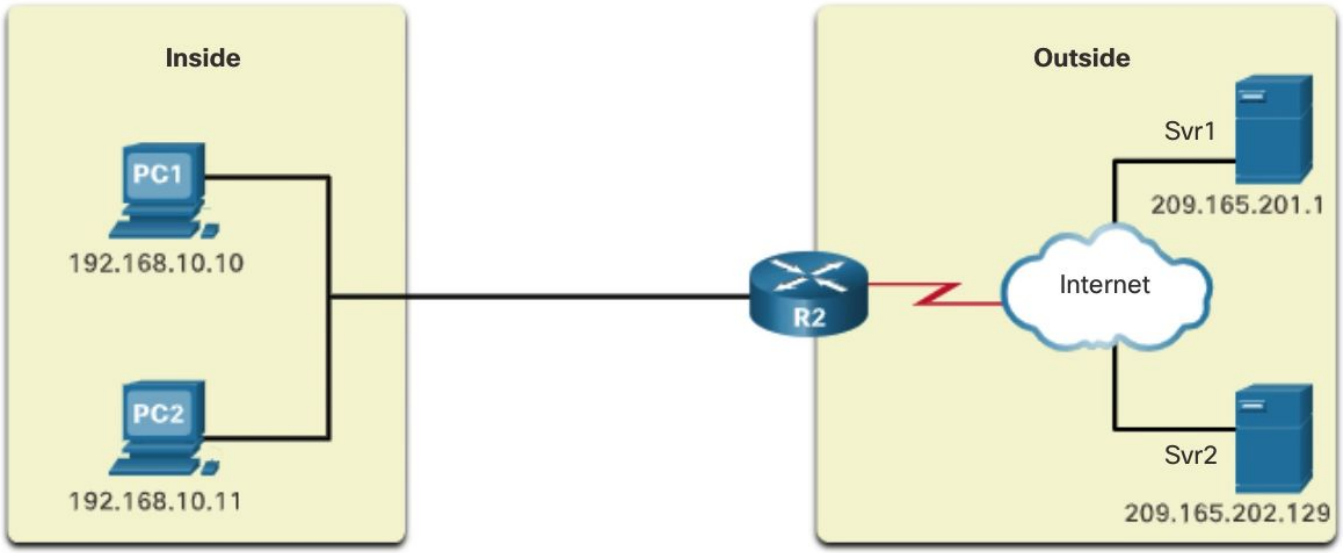
Port Address Translation (PAT)

- Port Address Translation (PAT), also known as NAT overload, maps multiple private IPv4 addresses to a single public IPv4 address or a few addresses.
- This is what most home routers do.
- The ISP assigns one address to the router, yet several members of the household can simultaneously access the internet.
- With PAT, multiple addresses can be mapped to one or to a few addresses, because each private address is also tracked by a port number.
- When a device initiates a TCP/IP session, it generates a TCP or UDP source port value to uniquely identify the session.
- When the NAT router receives a packet from the client, it uses its source port number to uniquely identify the specific NAT translation.

Port Address Translation (PAT) (cont.)

- PAT ensures that devices use a different TCP port number for each session with a server on the internet.
- When a response comes back from the server, the source port number, which becomes the destination port number on the return trip, determines to which device the router forwards the packets.
- The PAT process also validates that the incoming packets were requested, thus adding a degree of security to the session.
- PAT adds unique source port numbers to the inside global address to distinguish between translations.

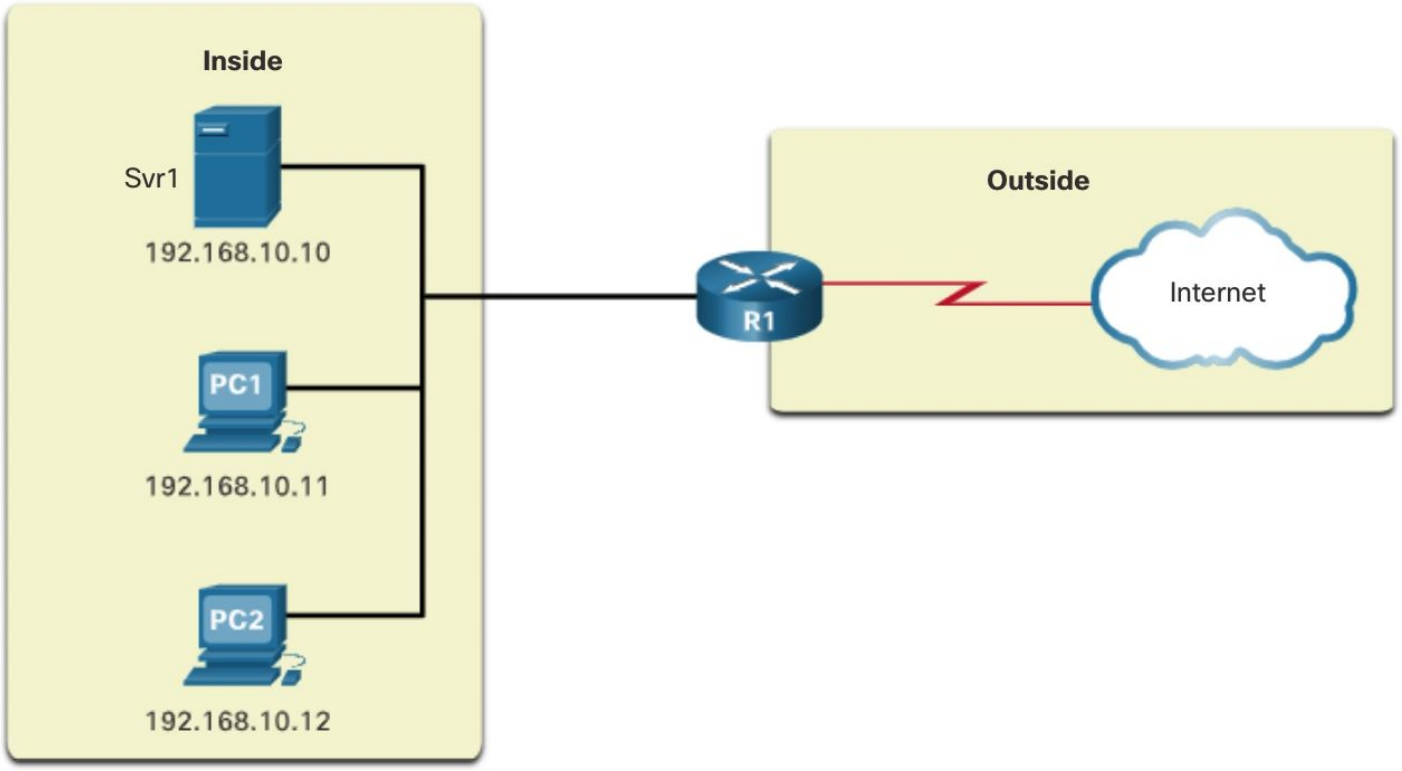
How PAT Works



Next Available Port

- In the previous example, the client port numbers, 1331 and 1555, did not change at the NAT-enabled router. This is not a very likely scenario, because there is a good chance that these port numbers may have already been attached to other active sessions.
- PAT attempts to preserve the original source port.
- However, if the original source port is already used, PAT assigns the first available port number starting from the beginning of the appropriate port group 0-511, 512-1,023, or 1,024-65,535.
- When there are no more ports available and there is more than one external address in the address pool, PAT moves to the next address to try to allocate the original source port.
- This process continues until there are no more available ports or external IPv4 addresses.

Next Available Port (cont.)



NAT Tables

- In the table below, four hosts on the internal network are communicating to the outside network.
- The **left column** lists the addresses in the **global address pool** that are used by NAT to translate the Inside Local address of each host.
- Note the one-to-one relationship of Inside Global addresses to Inside Local addresses for each of the four hosts accessing the outside network.
- With NAT, an Inside Global address is needed for each host that needs to connect to the outside network.

Inside Global Address	Inside Local Address
209.165.200.226	192.168.10.10
209.165.200.227	192.168.10.11
209.165.200.228	192.168.10.12
209.165.200.229	192.168.10.13

Note: NAT forwards the incoming return packets to the original inside host by referring to the table and translating the Inside Global address back to the corresponding Inside Local address of the host.

PAT Table

- NAT only modifies the IPv4 addresses, PAT modifies both the IPv4 address and the port number.
- With PAT, there is generally only one, or very few, publicly exposed IPv4 addresses.
- PAT uses the Layer 4 port number to track the conversations of the four hosts.

Inside Global Address	Inside Local Address
209.165.200.226:2031	192.168.10.10:2031
209.165.200.226:1506	192.168.10.11:1506
209.165.200.226:1131	192.168.10.12:1131
209.165.200.226:1718	192.168.10.13:1718

NAT and PAT Comparison

The table provides a summary of the differences between NAT and PAT.

NAT	PAT
One-to-one mapping between Inside Local and Inside Global addresses.	One Inside Global address can be mapped to many Inside Local addresses.
Uses only IPv4 addresses in translation process.	Uses IPv4 addresses and TCP or UDP source port numbers in translation process.
A unique Inside Global address is required for each inside host accessing the outside network.	A single unique Inside Global address can be shared by many inside hosts accessing the outside network.

Advantages of NAT

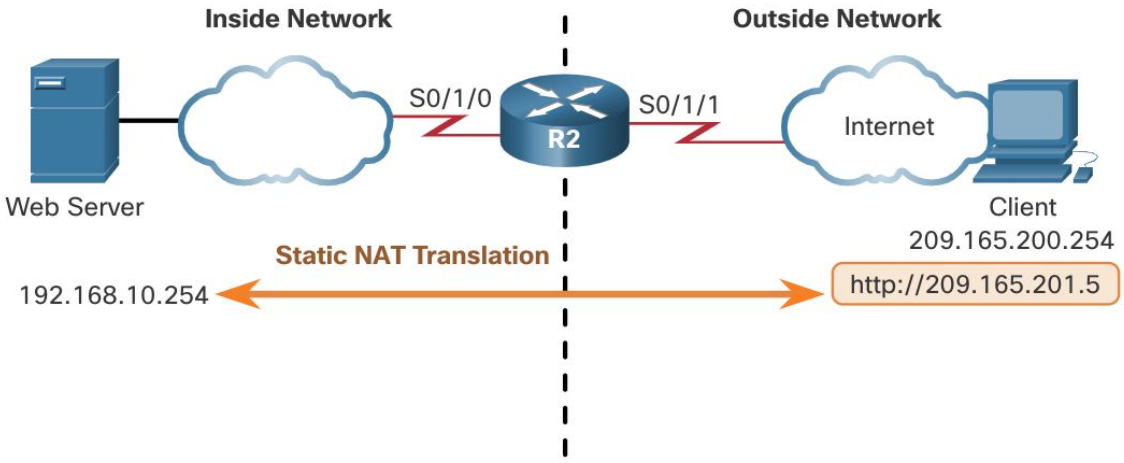
- NAT solves our problem of not having enough IPv4 addresses.
- With NAT overload (PAT), internal hosts can share a single public IPv4 address for all external communications.
- In this type of configuration, very few external addresses are required to support many internal hosts.
- NAT increases the flexibility of connections to the public network. Multiple pools, backup pools, and load-balancing pools can be implemented to ensure reliable public network connections.
- Hiding the IPv4 addresses of users and other devices can be a security feature.

Disadvantages of NAT

- The fact that hosts on the internet appear to communicate directly with the NAT-enabled device, rather than with the actual host inside the private network.
- NAT increases forwarding delays because the translation of each IPv4 address within the packet headers takes time.
- The router must look at every packet to decide whether it needs translation.
- The router must alter the IPv4 header which must be recalculated each time a translation is made, and possibly alter the TCP or UDP header.
- Remaining packets go through the fast-switched path if a cache entry exists; otherwise, they too are delayed.
- End-to-end addressing is lost. This is known as the end-to-end principle.
- Using NAT also complicates the use of tunneling protocols, such as IPsec, because NAT modifies values in the headers, causing integrity checks to fail.

Static NAT Scenario

- Static NAT is a one-to-one mapping between an inside address and an outside address.
- Static NAT allows external devices to initiate connections to internal devices using the statically assigned public address.
- For instance, an internal web server may be mapped to a specific inside global address so that it is accessible from outside networks.
- Router R2 is configured with static NAT to allow devices on the outside network (internet) to access the web server.
- The client on the outside network accesses the web server using a public IPv4 address.



Configure Static NAT

- There are two basic tasks when configuring static NAT translations:
- Step 1. The first task is to create a mapping between the inside local address and the inside global addresses.
- Exp: The 192.168.10.254 inside local address and the 209.165.201.5 inside global address in the figure are configured as a static NAT translation.

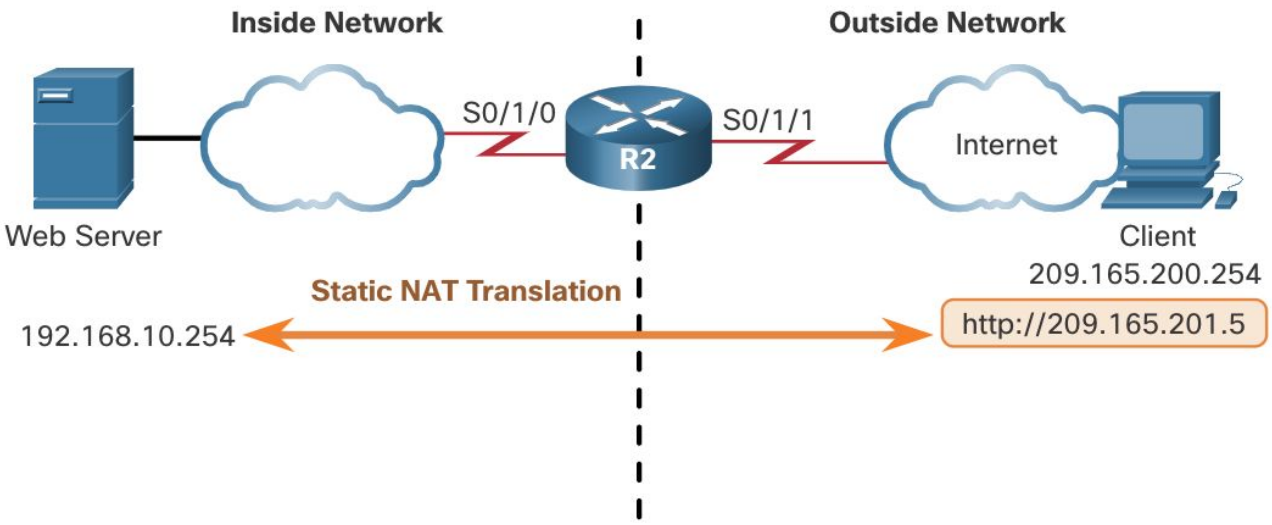
```
R2(config)# ip nat inside source static 192.168.10.254 209.165.201.5
```

- Step 2. After the mapping is configured, the interfaces participating in the translation are configured as inside or outside relative to NAT.
- In the example, the R2 Serial 0/1/0 interface is an inside interface and Serial 0/1/1 is an outside interface.

```
R2(config)# interface serial 0/1/0
R2(config-if)# ip address 192.168.1.2 255.255.255.252
R2(config-if)# ip nat inside
R2(config-if)# exit
R2(config)# interface serial 0/1/1
R2(config-if)# ip address 209.165.200.1 255.255.255.252
R2(config-if)# ip nat outside
```

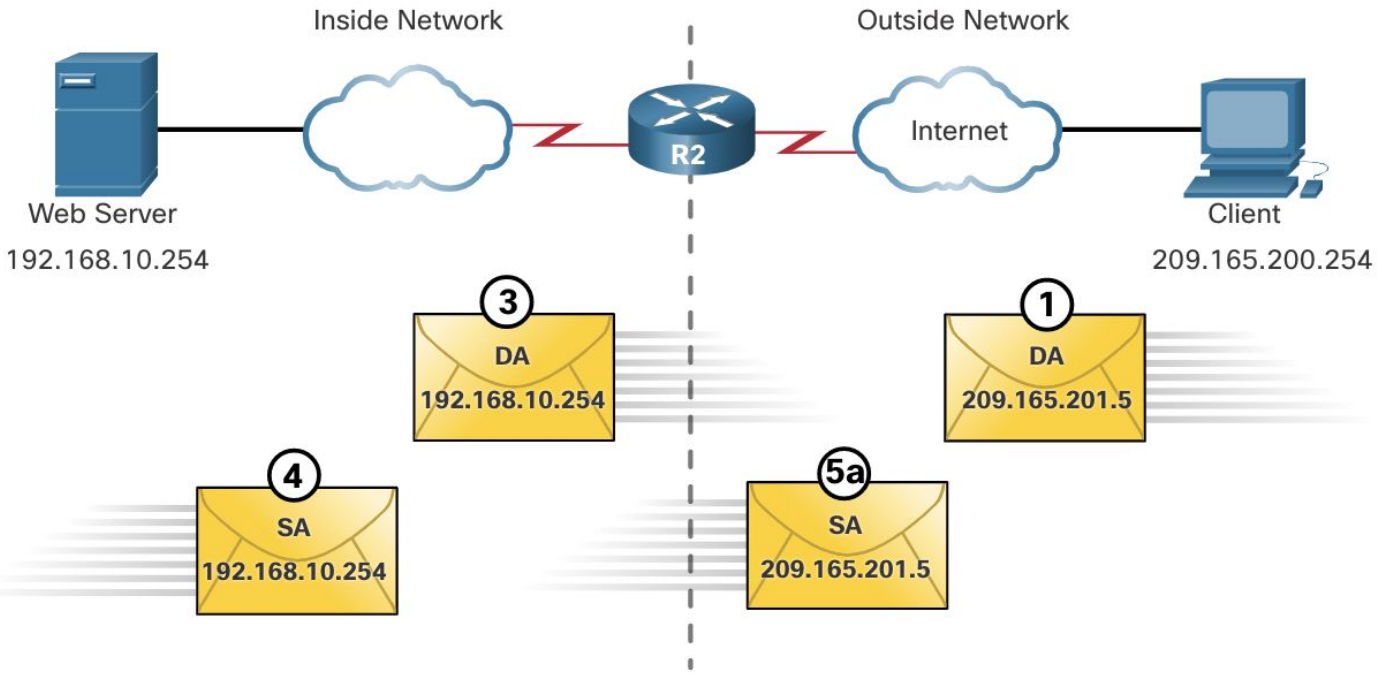
Configure Static NAT (cont.)

- In the second configuration, packets arriving on the inside interface of R2 (Serial 0/1/0) from the configured inside local IPv4 address (192.168.10.254) are translated and then forwarded towards the outside network.
- Packets arriving on the outside interface of R2 (Serial 0/1/1), that are addressed to the configured inside global IPv4 address (209.165.201.5), are translated to the inside local address (192.168.10.254) and then forwarded to the inside network.



NAT Translation Process

- Static translations are used when clients on the outside network (internet) need to reach servers on the inside (internal) network.



	Inside Local Address	Inside Global Address	Outside Local Address	Outside Global Address
2 5b	192.168.10.254	209.165.201.5	209.165.200.254	209.165.200.254

NAT Translation Process (cont.)

1. The client wants to open a connection to the web server. The client sends a packet to the web server using the public IPv4 destination address of 209.165.201.5. This is the inside global address of the web server.
2. The first packet that R2 receives from the client on its NAT outside interface causes R2 to check its NAT table. The destination IPv4 address of 209.165.201.5 is located in the NAT table and is translated to 192.168.10.254.
3. R2 replaces the inside global address of 209.165.201.5 with the inside local address of 192.168.10.254. R2 then forwards the packet towards the web server.
4. The web server receives the packet and responds to the client using the inside local address, 192.168.10.254 as the source address of the response packet.
5. (a) R2 receives the packet from the web server on its NAT inside interface with source address of the inside local address of the web server, 192.168.10.254.

   (b) R2 checks the NAT table for a translation for the inside local address. The address is found in the NAT table. R2 translates the source address 192.168.10.254 to the inside global address of 209.165.201.5 and forwards the packet toward the client.
6. The client receives the packet and continues the conversation. The NAT router performs Steps 2 to 5b for each packet.

Analyzing Static NAT

- To verify NAT operation **show ip nat translations** command is used.
- This command shows active NAT translations.

```
R2# show ip nat translations
Pro  Inside global       Inside local       Outside local      Outside global
---  209.165.201.5       192.168.10.254     ---                ---
Total number of translations: 1
```

- Another useful command is **show ip nat statistics**, which displays information about the total number of active translations, NAT configuration parameters, the number of addresses in the pool, and the number of addresses that have been allocated.
- To verify that the NAT translation is working, it is best to clear statistics from any past translations using the **clear ip nat statistics** command before testing.

```
R2# show ip nat statistics
Total active translations: 1 (1 static, 0 dynamic; 0 extended)
Outside interfaces:
  Serial0/1/1
Inside interfaces:
  Serial0/1/0
Hits: 0  Misses: 0
(output omitted)
```

Dynamic NAT Scenario

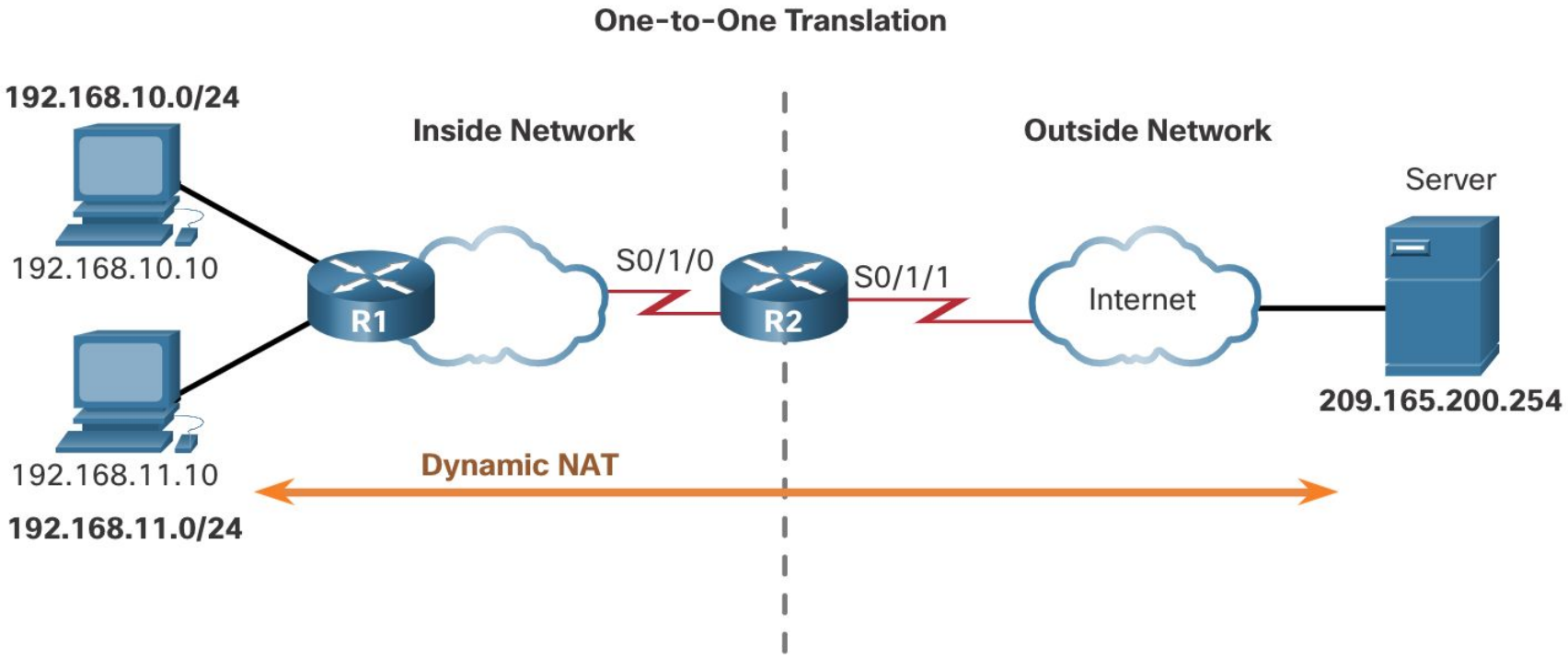
- Static NAT provides a permanent mapping between an inside local address and an inside global address, dynamic NAT automatically maps inside local addresses to inside global addresses.
- These inside global addresses are typically public IPv4 addresses.
- Dynamic NAT, like static NAT, requires the configuration of the inside and outside interfaces participating in NAT with the **ip nat inside** and **ip nat outside** interface configuration commands.
- However, where static NAT creates a permanent mapping to a single address, dynamic NAT uses a pool of addresses.

Dynamic NAT Scenario (cont.)

- The pool of public IPv4 addresses (inside global address pool) is available to any device on the inside network on a first-come first-served basis.
- With dynamic NAT, a single inside address is translated to a single outside address.
- With this type of translation there must be enough addresses in the pool to accommodate all the inside devices needing concurrent access to the outside network.
- If all addresses in the pool are in use, a device must wait for an available address before it can access the outside network.

Dynamic NAT Scenario (cont.)

- Attached to router R1 are two LANs, 192.168.10.0/24 and 192.168.11.0/24.
- Router R2, the border router, is configured for dynamic NAT using a pool of public IPv4 addresses 209.165.200.226 through 209.165.200.240.



Configuring Dynamic NAT

Step 1: Define the pool of addresses that will be used for translation using the **ip nat pool** command. The addresses are defined by indicating the starting IPv4 address and the ending IPv4 address of the pool.
The **netmask** or **prefix-length** keyword indicates which address bits belong to the network and which bits belong to the host for that range of addresses.
In the scenario, define a pool of public IPv4 addresses under the pool name NAT-POOL1.

```
R2(config)# ip nat pool NAT-POOL1 209.165.200.226 209.165.200.240 netmask 255.255.255.224
```

Step 2: Configure a standard ACL to identify (permit) only those addresses that are to be translated. An ACL that is too permissive can lead to unpredictable results. Remember there is an implicit **deny all** statement at the end of each ACL.
Below are defined the addresses which are eligible to be translated.

```
R2(config)# access-list 1 permit 192.168.0.0 0.0.255.255
```

Step 3: Bind the ACL to the pool, using the following command syntax:
Router(config)# **ip nat inside source list** {*access-list-number* | *access-list-name*} **pool** *pool-name*
This configuration is used by the router to identify which devices (**list**) receive which addresses (**pool**).
In the scenario, bind NAT-POOL1 with ACL 1.

```
R2(config)# ip nat inside source list 1 pool NAT-POOL1
```

Configuring Dynamic NAT (cont.)

Step 4: Identify which interfaces are inside, in relation to NAT; this will be any interface that connects to the inside network.
In the scenario, interface serial 0/1/0 is identified as an inside NAT interface.

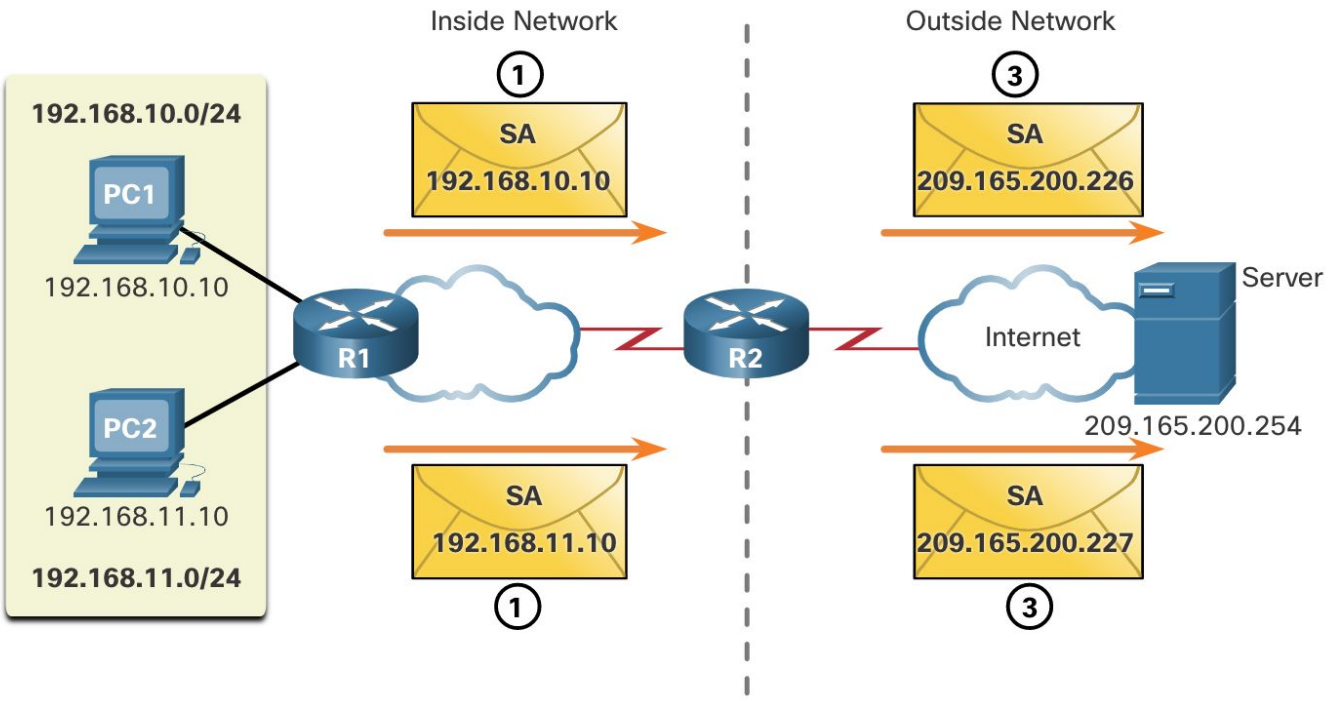
```
R2(config)# interface serial 0/1/0
R2(config-if)# ip nat inside
```

Step 5: Identify which interfaces are outside, in relation to NAT; this will be any interface that connects to the outside network.
In the scenario, interface serial 0/1/1 is identified as an inside NAT interface.

```
R2(config)# interface serial 0/1/1
R2(config-if)# ip nat outside
```

Analyzing Dynamic NAT – Inside to Outside

- The figure show the dynamic NAT translation process between two clients and the web server as well as the traffic flow from the **inside** network to the **outside**.



IPv4 NAT Pool	
Inside Local Address Pool	**Inside Global Address**
192.168.10.10	209.165.200.226
192.168.11.10	209.165.200.227

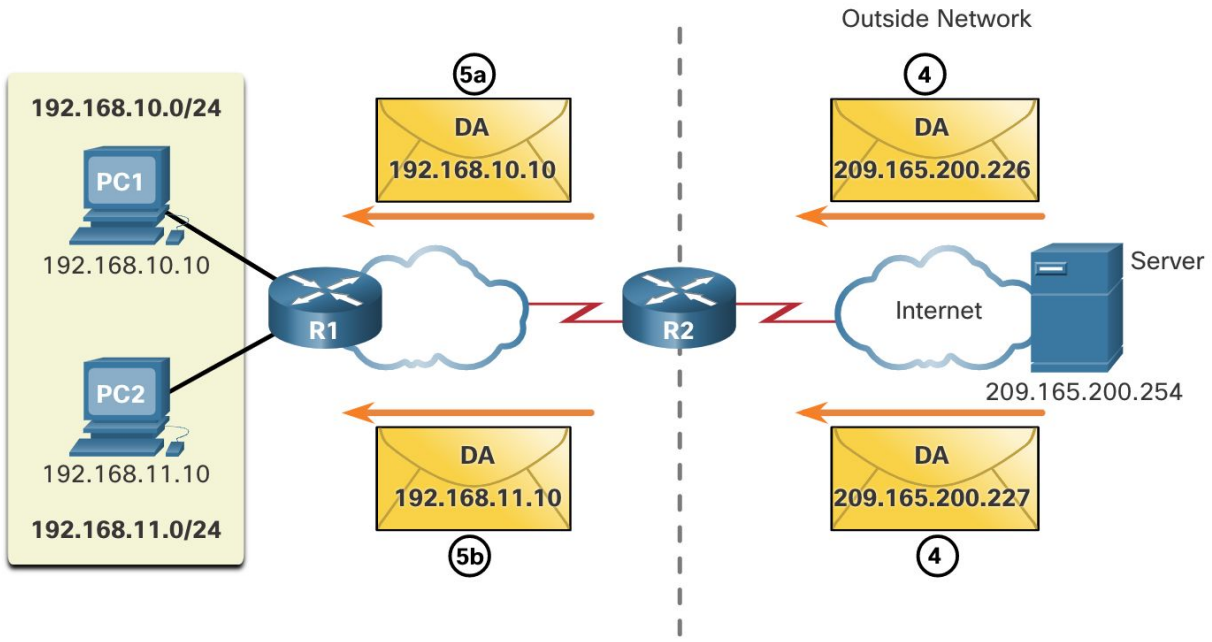
2
2

Analyzing Dynamic NAT – Inside to Outside (cont.)

1. The hosts with the source IPv4 addresses of 192.168.10.10 (PC1) and 192.168.11.10 (PC2) send packets requesting a connection to the server at the public IPv4 address 209.165.200.254.
2. R2 receives the first packet from host 192.168.10.10. Because this packet was received on an interface configured as an inside NAT interface, R2 checks the NAT configuration to determine if this packet should be translated. The ACL permits this packet, so R2 will translate the packet. R2 checks its NAT table. Because there is no current translation entry for this IPv4 address, R2 determines that the source address 192.168.10.10 must be translated. R2 selects an available global address from the dynamic address pool and creates a translation entry, 209.165.200.226. The original source IPv4 address 192.168.10.10 is the inside local address and the translated address is the inside global address 209.165.200.226 in the NAT table. For the second host, 192.168.11.10, R2 repeats the procedure, selects the next available global address from the dynamic address pool, and creates a second translation entry, 209.165.200.227.
3. R2 replaces the inside local source address of PC1, 192.168.10.10, with the translated inside global address of 209.165.200.226 and forwards the packet. The same process occurs for the packet from PC2 using the translated address of 209.165.200.227.

Analyze Dynamic NAT - Outside to Inside

The figure below illustrates the remainder of the traffic flow between the clients and the server from the **outside** to the **inside** direction.



IPv4 NAT Pool	
Inside Local Address Pool	**Inside Global Address**
192.168.10.10	209.165.200.226
192.168.11.10	209.165.200.227

5a
5b

Analyze Dynamic NAT - Outside to Inside (cont.)

4. The server receives the packet from PC1 and responds using the IPv4 destination address of 209.165.200.226. When the server receives the second packet, it responds to PC2 using the IPv4 destination address of 209.165.200.227.
5. (a) When R2 receives the packet with the destination IPv4 address of 209.165.200.226; it performs a NAT table lookup. Using the mapping from the table, R2 translates the address back to the inside local address 192.168.10.10 and forwards the packet toward PC1.

   (b) When R2 receives the packet with the destination IPv4 address of 209.165.200.227; it performs a NAT table lookup. Using the mapping from the table, R2 translates the address back to the inside local address 192.168.11.10 and forwards the packet toward PC2.
6. PC1 at 192.168.10.10 and PC2 at 192.168.11.10 receive the packets and continue the conversation. The router performs Steps 2 to 5 for each packet.

Verifying DNAT

- The output of the **show ip nat translations** command displays all static translations that have been configured and any dynamic translations that have been created by traffic.

```
R2# show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
--- 209.165.200.228    192.168.10.10      ---                ---
--- 209.165.200.229    192.168.11.10      ---                ---
R2#
```

- Adding the **verbose** keyword displays additional information about each translation, including how long ago the entry was created and used.

```
R2# show ip nat translation verbose
Pro Inside global      Inside local       Outside local      Outside global
tcp 209.165.200.228    192.168.10.10      ---                ---
    create 00:02:11, use 00:02:11 timeout:86400000, left 23:57:48, Map-Id(In): 1,
    flags:
none, use_count: 0, entry-id: 10, lc_entries: 0
tcp 209.165.200.229    192.168.11.10      ---                ---
    create 00:02:10, use 00:02:10 timeout:86400000, left 23:57:49, Map-Id(In): 1,
    flags:
none, use_count: 0, entry-id: 12, lc_entries: 0
R2#
```

Verifying DNAT (cont.)

- By default, translation entries time out after 24 hours, unless the timers have been reconfigured with the **ip nat translation timeout** *timeout-seconds* command in global configuration mode.

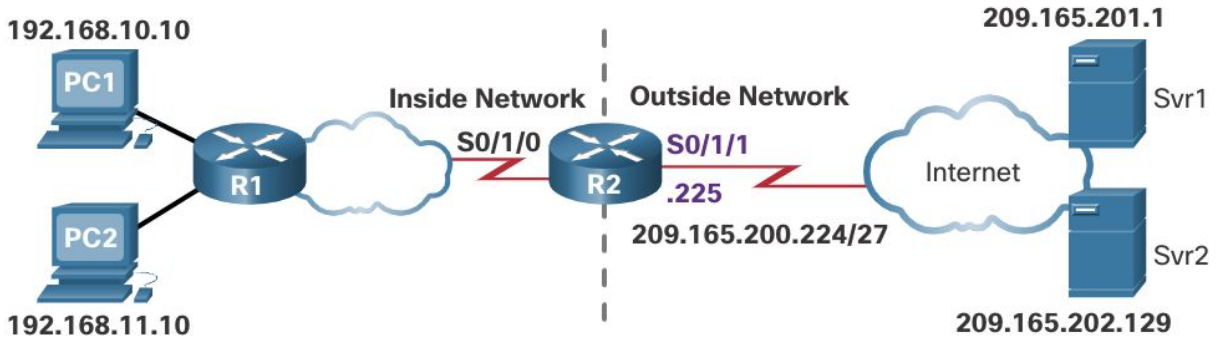
```
R2# clear ip nat translation *
R2# show ip nat translation
```

- To clear dynamic entries before the timeout has expired, use the **clear ip nat translation** privileged EXEC mode command as shown.

Command	Description
clear ip nat translation *	Clears all dynamic address translation entries from the NAT translation table.
clear ip nat translation inside*global-ip local-ip* [**outside** *local-ip global-ip*]	Clears a simple dynamic translation entry containing an inside translation or both inside and outside translation.
clear ip nat translation *protocol***inside***global-ip global-port local-ip local-port* [**outside***local-ip local-port global-ip global-port*]	Clears an extended dynamic translation entry.

PAT Scenario

- There are two ways to configure PAT, depending on how the ISP allocates public IPv4 addresses.
  a) the ISP allocates a single public IPv4 address that is required for the organization to connect to the ISP.
  b) the ISP allocates more than one public IPv4 address to the organization.



NAT Table			
Inside Local Address	**Inside Global Address**	**Outside Global Address**	**Outside Local Address**
192.168.10.10:1444	209.165.200.225:1444	209.165.201.1:80	209.165.201.1:80
192.168.11.10:1444	209.165.200.225:1445	209.165.202.129:80	209.165.202.129:80

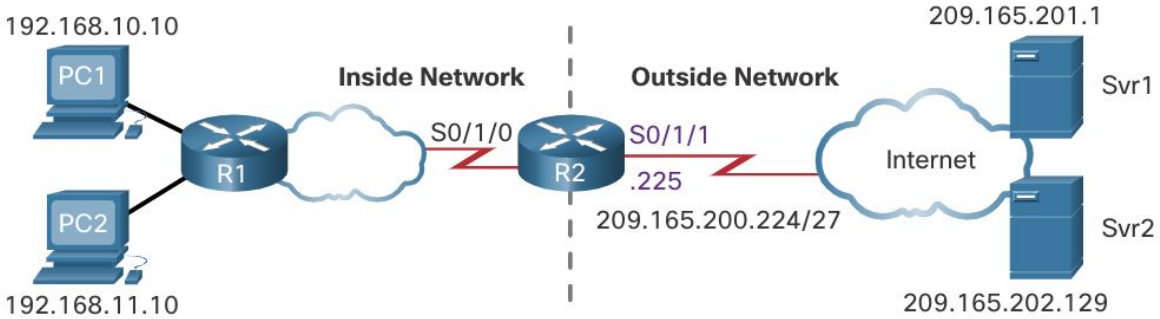
Configure PAT to Use a Single IPv4 Address

- To configure PAT to use a single IPv4 address, simply add the keyword **overload** to the **ip nat inside source** command.
- The rest of the configuration is the similar to static and dynamic NAT configuration **except** that with PAT, **multiple hosts** can use the **same public IPv4 address** to access the internet.
- In the example, all hosts from network **192.168.0.0/16** (matching ACL 1) that send traffic through router **R2** to the internet will be translated to IPv4 address **209.165.200.225** (IPv4 address of interface S0/1/1).
- The traffic flows will be identified by port numbers in the NAT table because the **overload** keyword is configured.

```
R2(config)# ip nat inside source list 1 interface serial 0/1/0 overload
R2(config)# access-list 1 permit 192.168.0.0 0.0.255.255
R2(config)# interface serial0/1/0
R2(config-if)# ip nat inside
R2(config-if)# exit
R2(config)# interface Serial0/1/1
R2(config-if)# ip nat outside
```

Configure PAT to Use an Address Pool

- An ISP may allocate more than one public IPv4 address to an organization. In this scenario the organization can configure PAT to use a pool of IPv4 public addresses for translation.
- If a site has been issued more than one public IPv4 address, these addresses can be part of a pool that is used by PAT.
- The **small pool** of addresses is shared among a **larger number of devices**, with **multiple hosts** using the **same public IPv4** address to access the internet.
- To configure PAT for a dynamic NAT address pool, simply add the keyword **overload** to the **ip nat inside source** command.



Configure PAT to Use an Address Pool (cont.)

- In the example, NAT-POOL2 is bound to an ACL to permit 192.168.0.0/16 to be translated.
- These hosts can share an IPv4 address from the pool because PAT is enabled with the keyword overload.

```
R2(config)# ip nat pool NAT-POOL2 209.165.200.226 209.165.200.240 netmask 255.255.255.224
R2(config)# access-list 1 permit 192.168.0.0 0.0.255.255
R2(config)# ip nat inside source list 1 pool NAT-POOL2 overload
R2(config)#
R2(config)# interface serial0/1/0
R2(config-if)# ip nat inside
R2(config-if)# exit
R2(config)# interface serial0/1/1
R2(config-if)# ip nat outside
R2(config-if)# end
R2#
```

COURSE: **NETWORK ADMINISTRATION AND MANAGEMENT**

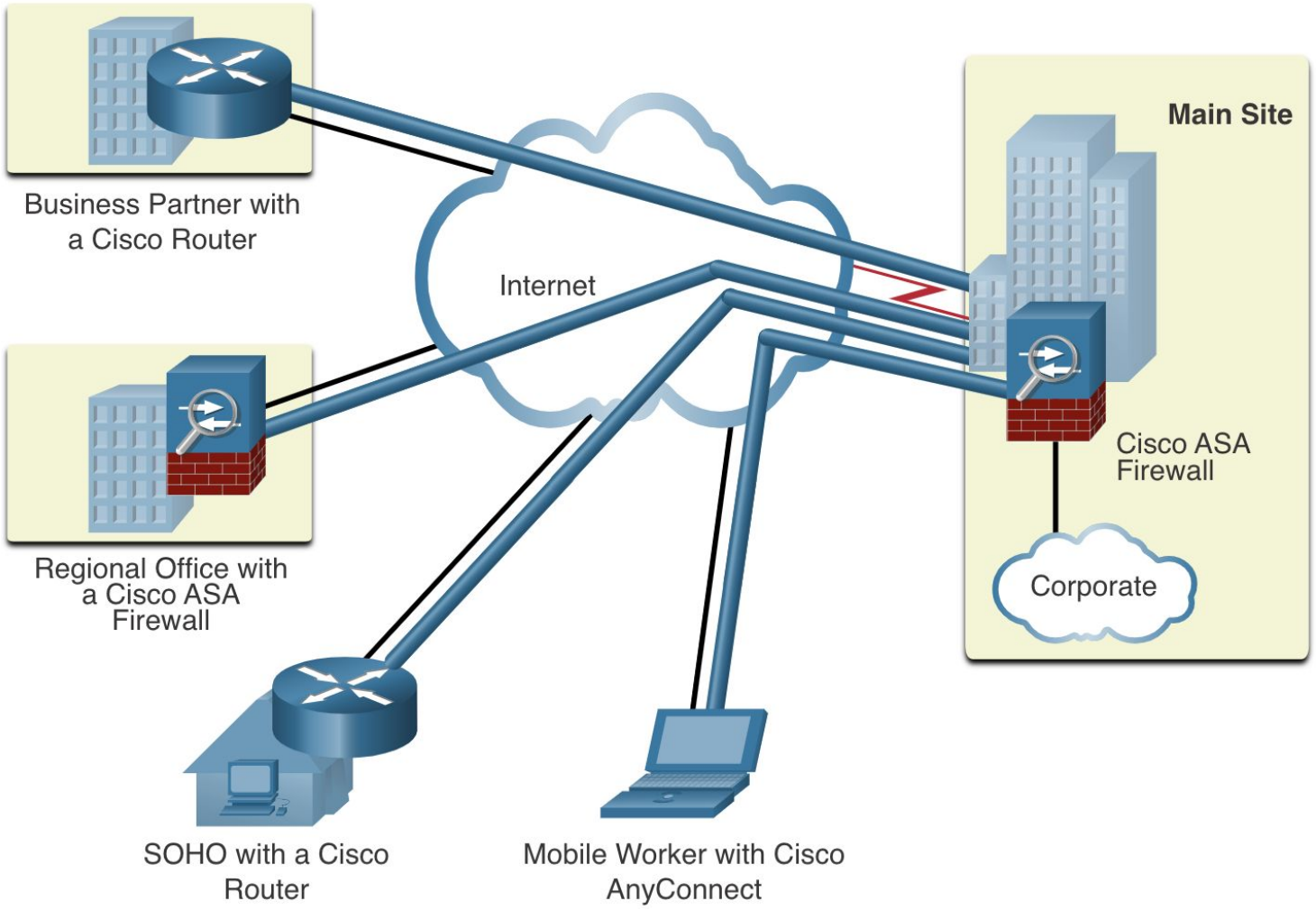
COURSE INSTRUCTOR: **MIRALDA CUKA, PHD**

Lecture 11
VPN and IPSec

Virtual Private Network

- To secure network traffic between sites and users, organizations use virtual private networks (VPNs) to create end-to-end private network connections.
- A VPN is virtual in that it carries information within a private network, but that information is actually transported over a public network.
- Traffic is encrypted to keep the data confidential while it is transported across the public network.
- The figure shows a collection of various types of VPNs managed by an enterprise's main site.
- The tunnel enables remote sites and users to access main site's network resources securely.

VPN Technology



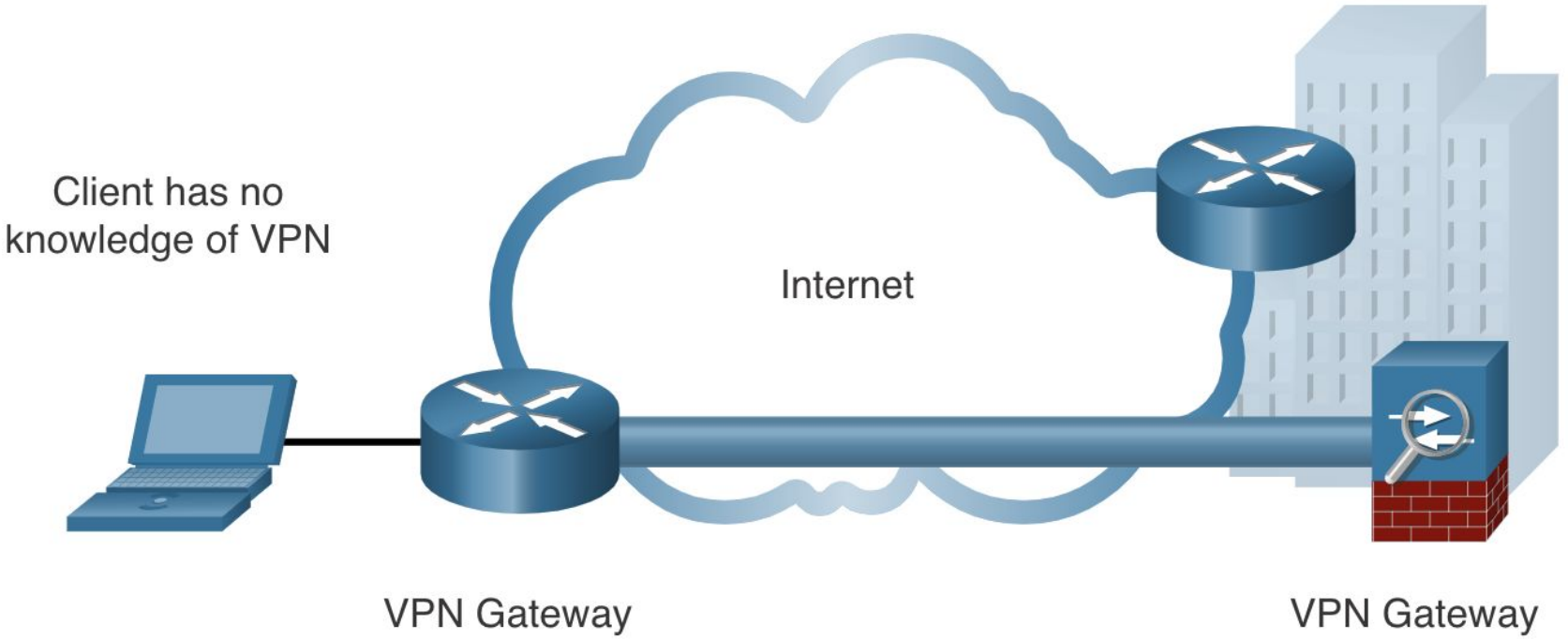
VPN Benefits

- To secure network traffic between sites VPNs support encryption features, such as:
  - Internet Protocol Security (IPsec).
  - Secure Sockets Layer (SSL) VPNs.
- Major benefits of VPNs are shown in the table.

Benefit	Description
Cost Savings	With the advent of cost-effective, high-bandwidth technologies, organizations can use VPNs to reduce their connectivity costs while simultaneously increasing remote connection bandwidth.
Security	VPNs provide the highest level of security available, by using advanced encryption and authentication protocols that protect data from unauthorized access.
Scalability	VPNs allow organizations to use the internet, making it easy to add new users without adding significant infrastructure.
Compatibility	VPNs can be implemented across a wide variety of WAN link options including all the popular broadband technologies. Remote workers can take advantage of these high-speed connections to gain secure access to their corporate networks.

Site-to-Site VPNs

- A site-to-site VPN is created when VPN terminating devices, also called VPN gateways, are preconfigured with information to establish a secure tunnel.
- VPN traffic is only encrypted between these devices. Internal hosts have no knowledge that a VPN is being used.



Remote Access VPN

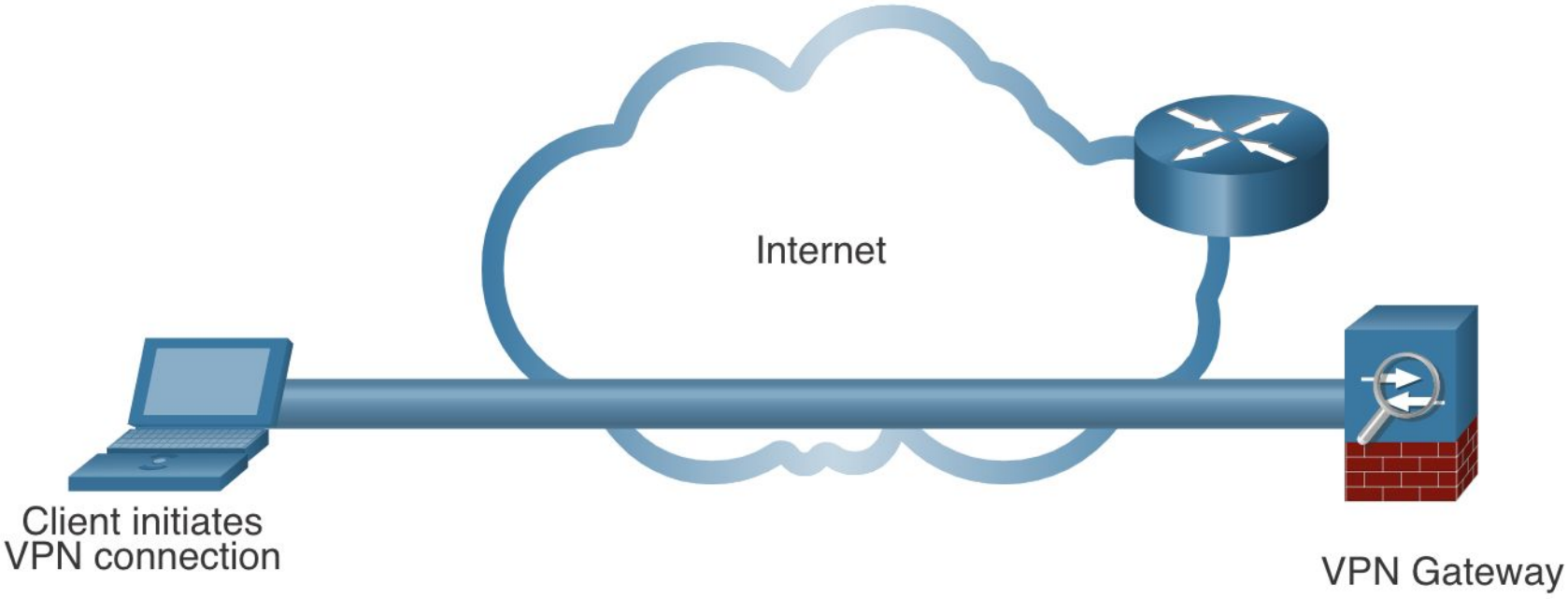
- Remote-access VPNs let remote and mobile users securely connect to the enterprise by creating an encrypted tunnel.
- Remote users can securely replicate their enterprise security access including email and network applications.
- Remote-access VPNs also allow contractors and partners to have limited access to the specific servers, web pages, or files as required without compromising network security.
- Remote-access VPNs are typically enabled dynamically by the user when required. Remote access VPNs can be created using either IPsec or SSL.
- A remote user must initiate a remote access VPN connection.
- There are two ways that a remote user can initiate a remote access VPN connection:
    - clientless VPN
    - client-based VPN

Remote Access VPN (cont.)

- Clientless VPN connection
  - The connection is secured using a web browser SSL connection.
  - SSL is mostly used to protect HTTP traffic (HTTPS) and email protocols such as IMAP and POP3.
  - For example, HTTPS is actually HTTP using an SSL tunnel. The SSL connection is first established, and then HTTP data is exchanged over the connection.
- Client-based VPN connection
  - VPN client software must be installed on the remote user's end device.
  - Users must initiate the VPN connection using the VPN client and then authenticate to the destination VPN gateway.
  - When remote users are authenticated, they have access to corporate files and applications. The VPN client software encrypts the traffic using IPsec or
  - SSL and forwards it over the internet to the destination VPN gateway.

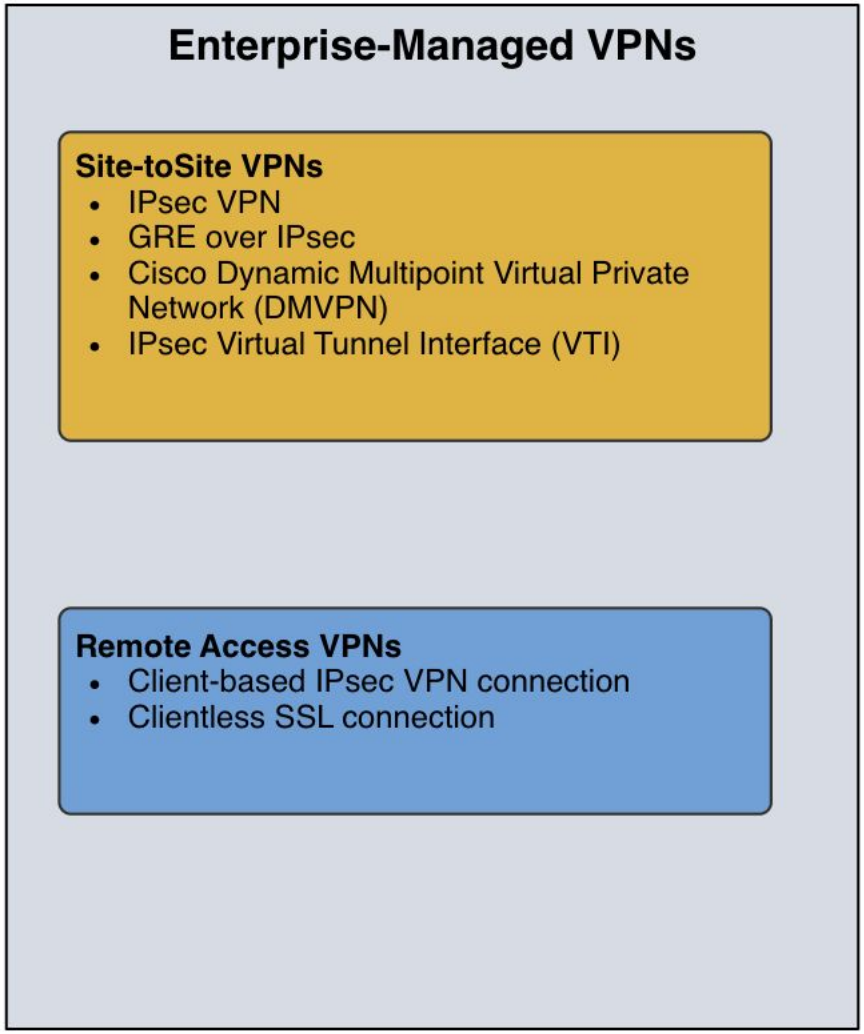
Remote-Access VPN

- A remote-access VPN is dynamically created to establish a secure connection between a client and a VPN terminating device.
- A remote access SSL VPN is used when you check your banking information online.



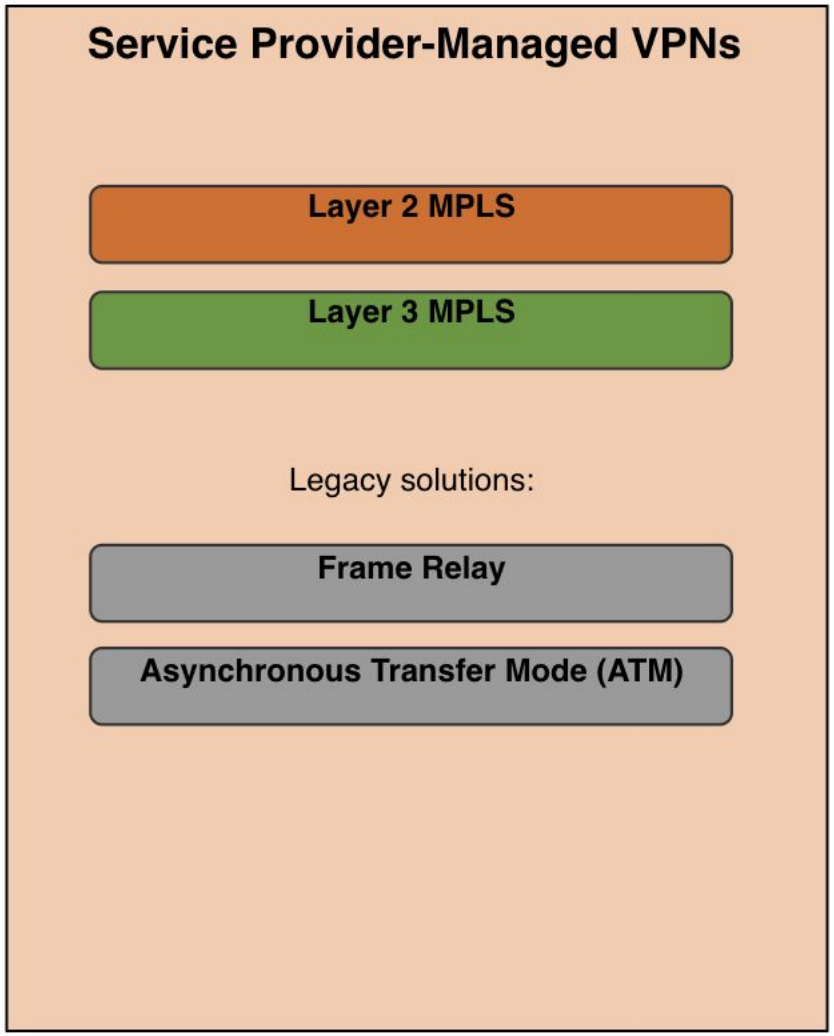
Enterprise VPNs

- Enterprise VPNs - Enterprise-managed VPNs are a common solution for securing enterprise traffic across the internet. Site-to-site and remote access VPNs are created and managed by the enterprise using both IPsec and SSL VPNs

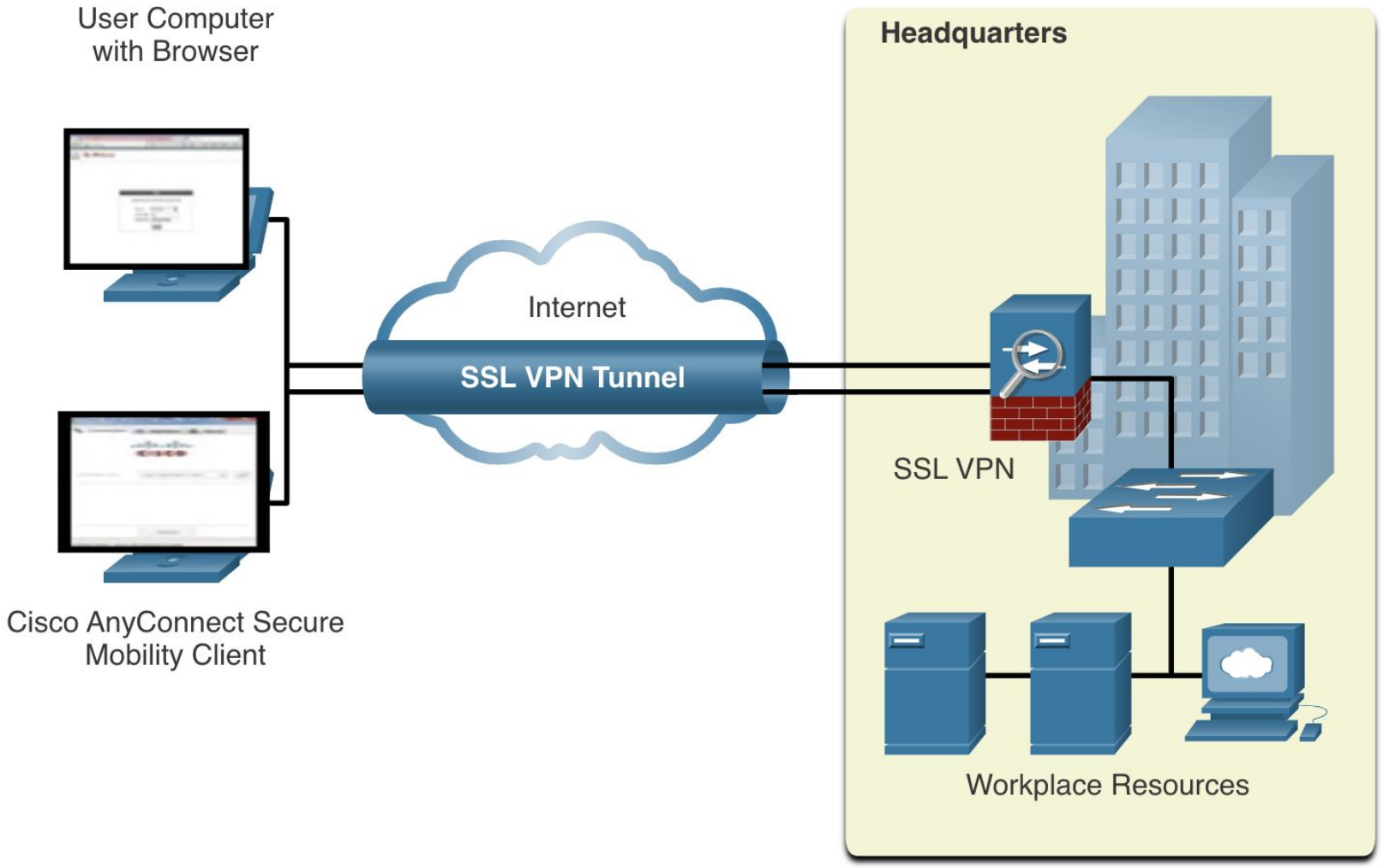


Service Provider VPNs

- Service Provider VPNs - Service provider-managed VPNs are created and managed over the provider network.
- The provider uses Multiprotocol Label Switching (MPLS) at Layer 2 or Layer 3 to create secure channels between an enterprise's sites.
- MPLS is a routing technology the provider uses to create virtual paths between sites.
- This effectively segregates the traffic from other customer traffic.



Remote Access VPN (cont.)



Remote-Access vs Site-to-site VPN

Key Differences:

- The purpose of a Site-to-Site VPN is to connect entire networks to each other, while a Remote Access VPN is used to connect individual devices to a network.
- In a Site-to-Site VPN, there is no need for individual users or devices to initiate the VPN connection as the connection is always on, while in a Remote Access VPN, individual users need to initiate the connection.
- Site-to-Site VPNs are ideal for connecting geographically dispersed offices, while Remote Access VPNs are ideal for employees who need to access their company's network while traveling or working remotely.
- In terms of security protocols, both types can utilize similar encryption methods, such as IPsec or SSL/TLS, depending on the specific implementation.

SSL and IPSec VPN Comparison

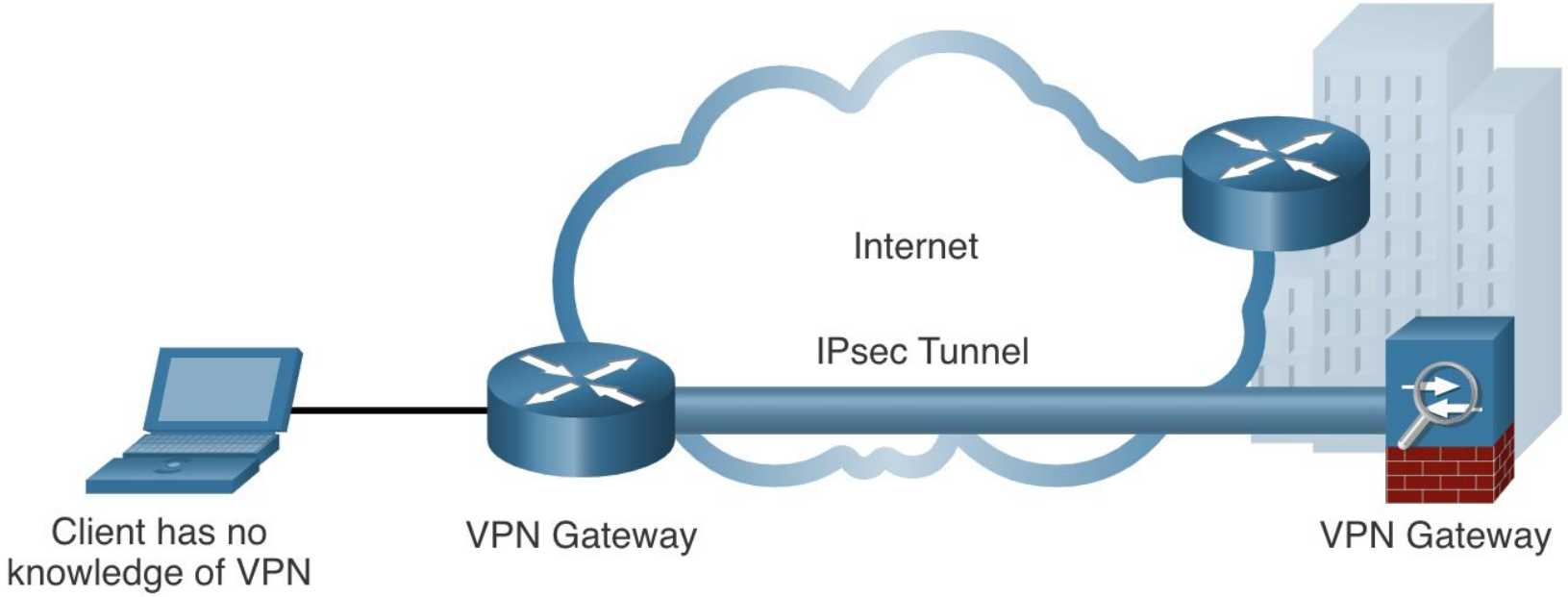
Feature	IPsec	SSL
Applications supported	**Extensive** - All IP-based applications are supported.	**Limited** - Only web-based applications and file sharing are supported.
Authentication strength	**Strong** - Uses two-way authentication with shared keys or digital certificates.	**Moderate** - Using one-way or two-way authentication.
Encryption strength	**Strong** - Uses key lengths from 56 bits to 256 bits.	**Moderate to strong** - With key lengths from 40 bits to 256 bits.
Connection complexity	**Medium** - Because it requires a VPN client pre-installed on a host.	**Low** - It only requires a web browser on a host.
Connection option	**Limited** - Only specific devices with specific configurations can connect.	**Extensive** - Any device with a web browser can connect.

Site-to-Site IPSec VPN

- Site-to-site VPNs are used to connect networks across another untrusted network such as the internet.
- In a site-to-site VPN, end hosts send and receive normal unencrypted TCP/IP traffic through a VPN terminating device.
- The VPN terminating is typically called a VPN gateway.
- A VPN gateway device could be a router or a firewall.
- A standalone firewall device can combine firewall, VPN concentrator, and intrusion prevention functionality into one software image.

Site-to-Site IPSec VPN (cont.)

- The VPN gateway **encapsulates** and **encrypts** outbound traffic.
- It then sends the traffic through a VPN tunnel over the internet to a VPN gateway at the target site.
- Upon receipt, the receiving VPN gateway strips the headers, decrypts the content, and relays the packet toward the target host inside its private network.
- Site-to-site VPNs are typically created and secured using IP security (IPsec).

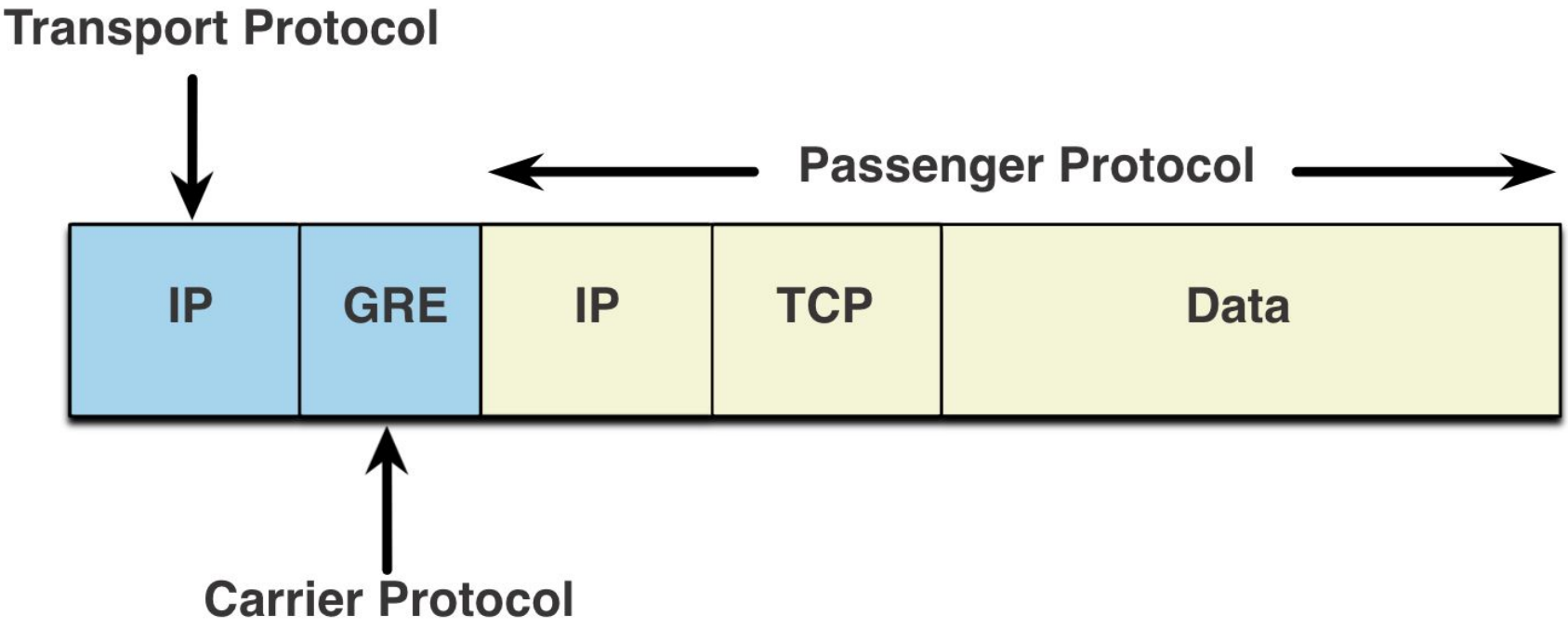


GRE over IPSec

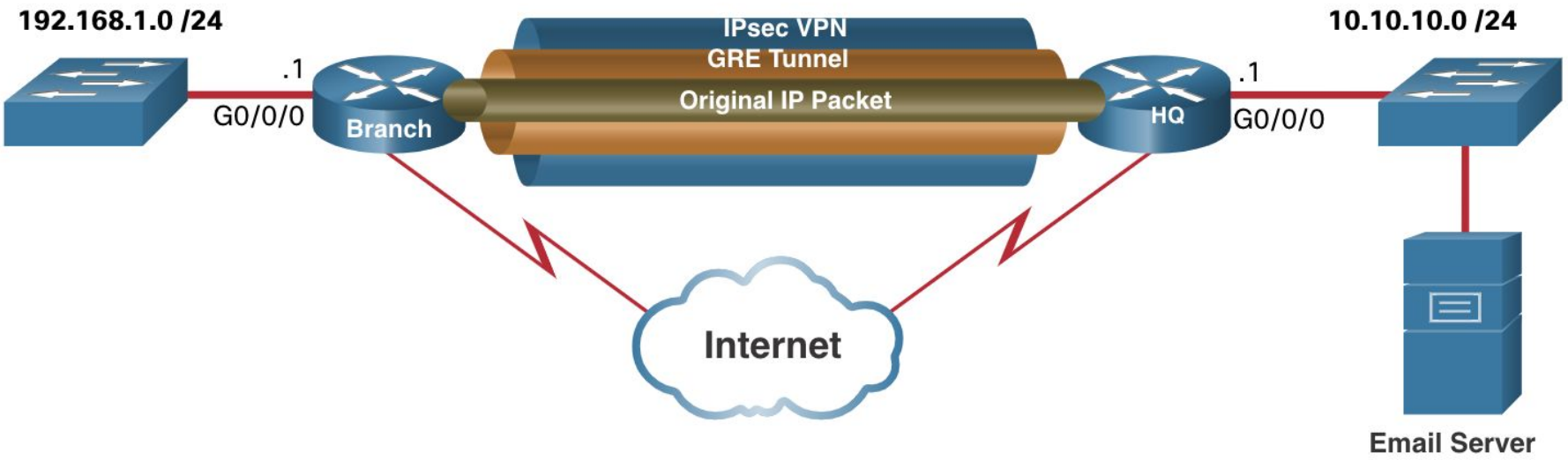
- Generic Routing Encapsulation (GRE) is a non-secure site-to-site VPN tunneling protocol.
- It supports **multicast** and **broadcast** traffic which may be necessary if the organization requires routing protocols to operate over a VPN.
- GRE does not by default support encryption; and therefore, it does not provide a secure VPN tunnel.
- A standard **IPsec VPN** (non-GRE) can only create **secure tunnels** for **unicast traffic**. Routing protocols will not exchange routing information over an IPsec VPN.
- To solve this problem, we can **encapsulate** routing protocol traffic using a **GRE packet**, and then **encapsulate the GRE packet** into an **IPsec packet** to forward it securely to the destination VPN gateway.

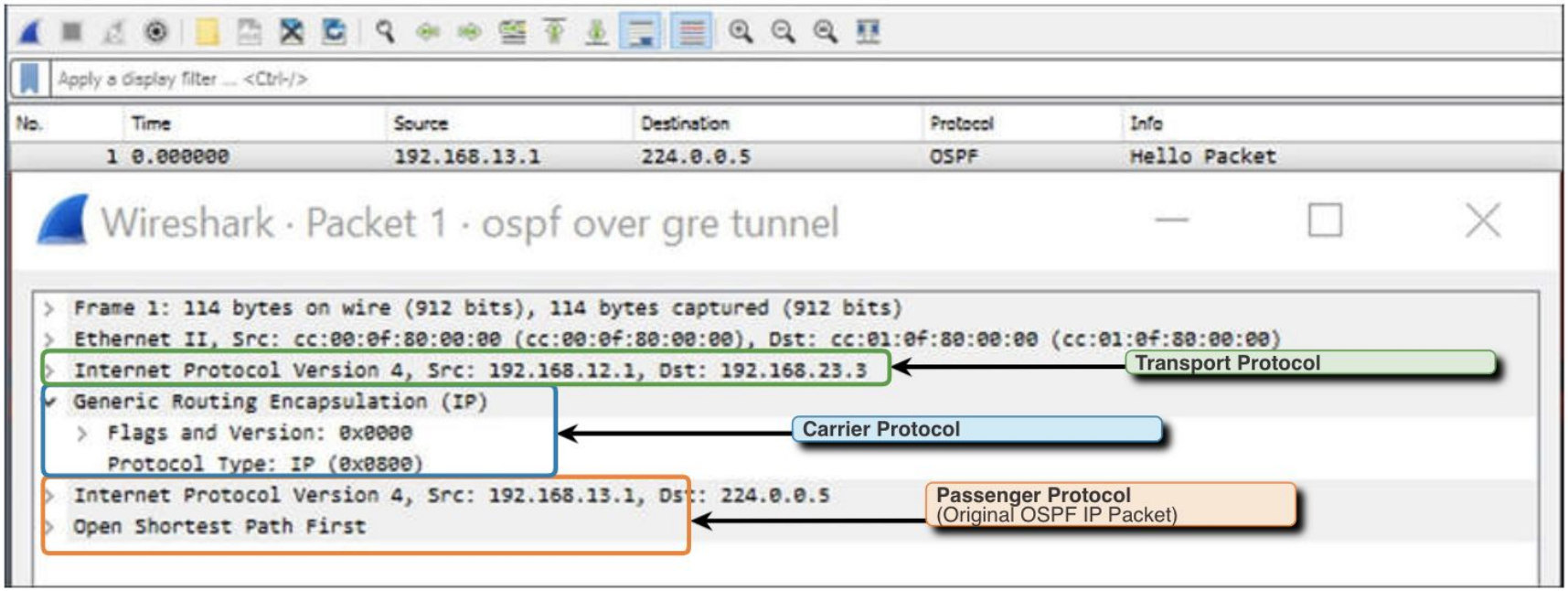
GRE over IPSec (cont.)

- Passenger protocol – This is the original packet that is to be encapsulated by GRE. It could be an IPv4 or IPv6 packet, a routing update, and more.
- Carrier protocol – GRE is the carrier protocol that encapsulates the original passenger packet.
- Transport protocol – This is the protocol that will actually be used to forward the packet. This could be IPv4 or IPv6.



GRE over IPSec (cont.)

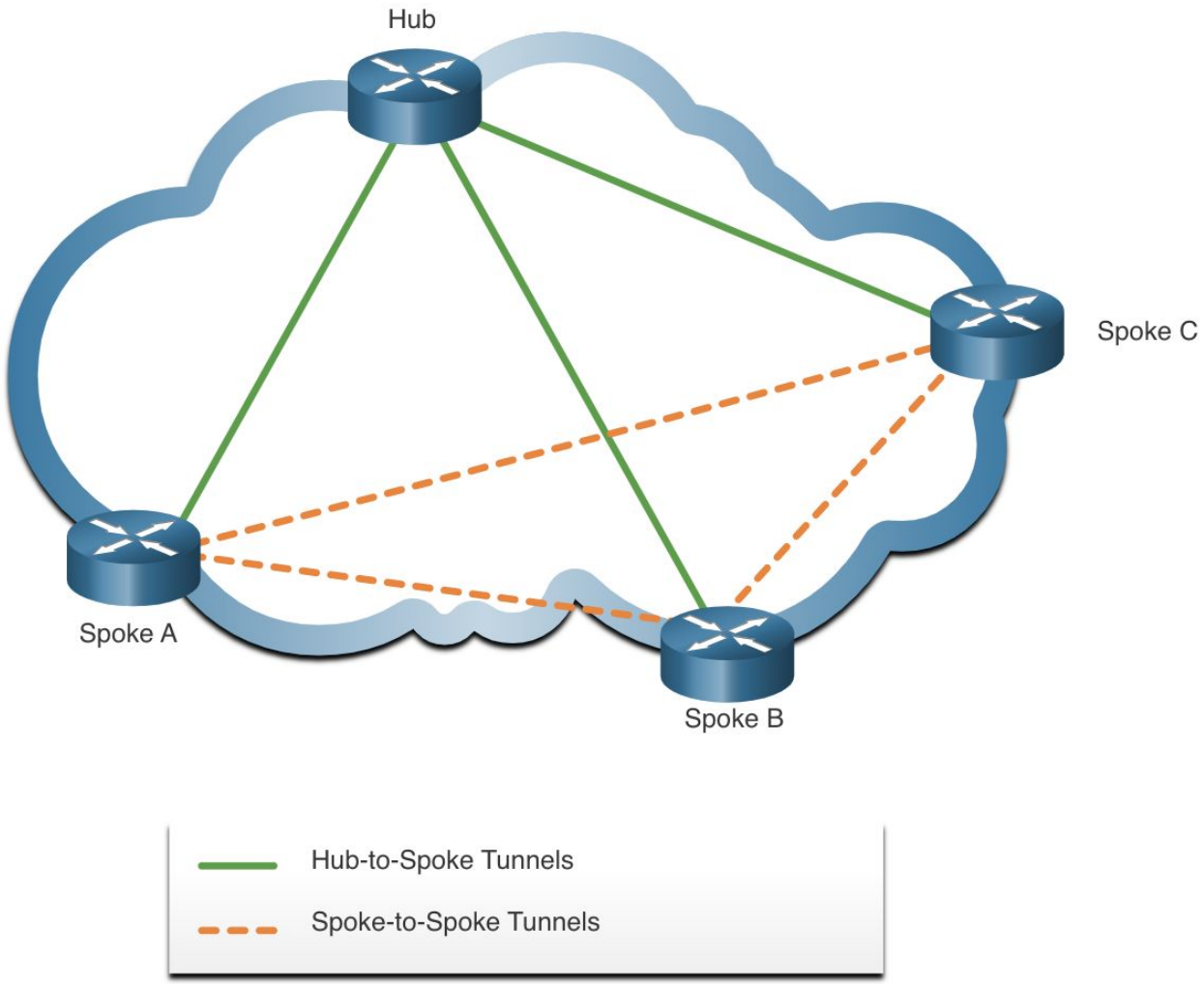




Dynamic Multipoint VPNs

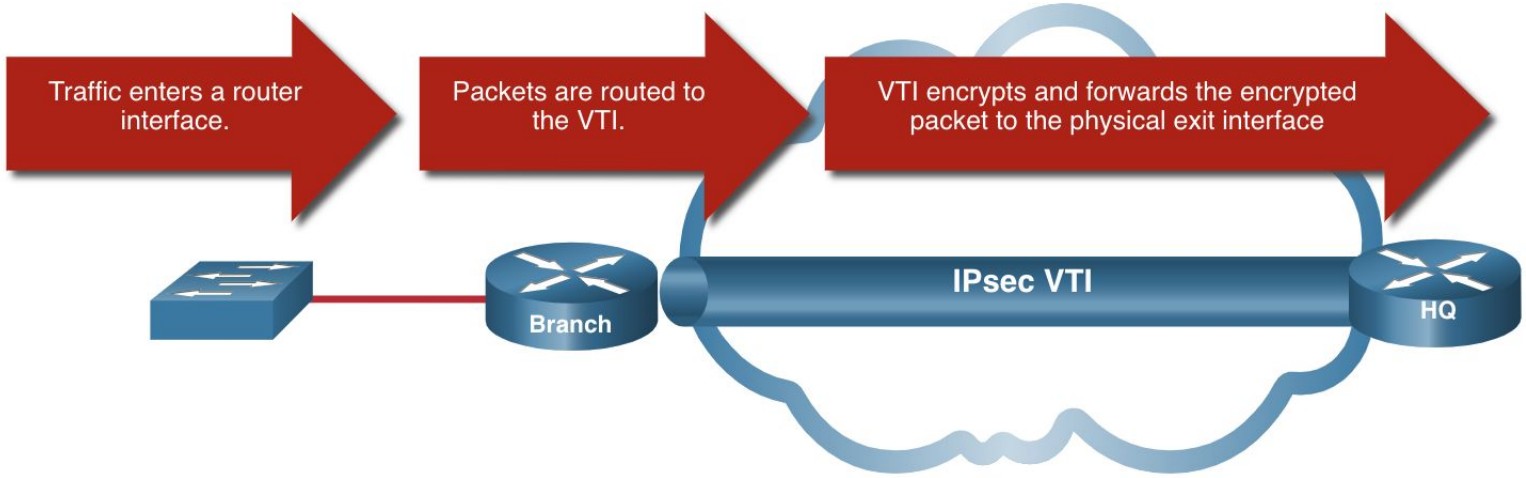
- Site-to-site IPsec VPNs and GRE over IPsec are adequate to use when there are only a few sites to securely interconnect.
- However, they are not sufficient when the enterprise adds many more sites. Each site would require static configurations to all other sites, or to a central site.
- Dynamic Multipoint VPN (DMVPN) is used for building multiple VPNs in an easy, dynamic, and scalable manner.
- Like other VPN types, DMVPN relies on IPsec to provide secure transport over public networks, such as the internet.
- DMVPN simplifies the VPN tunnel configuration and provides a flexible option to connect a central site with branch sites.

Dynamic Multipoint VPNs



IPsec Virtual Tunnel Interface

- IPsec Virtual Tunnel Interface (VTI) simplifies the configuration process required to support multiple sites and remote access.
- IPsec VTI configurations are applied to **a virtual interface** instead of static mapping the IPsec sessions to a physical interface.
- IPsec VTI is capable of sending and receiving both **IP unicast** and **multicast** encrypted traffic.
- Therefore, routing protocols are automatically supported without having to configure GRE tunnels.
- IPsec VTI can be configured between sites.



Service Provider MPLS VPNs

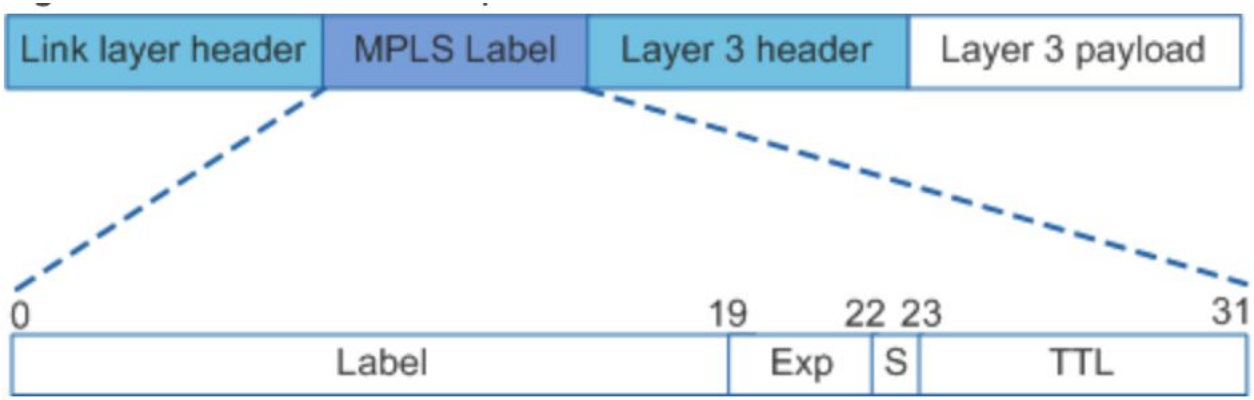
- Today, service providers use MPLS in their core network.
- Traffic is forwarded through the MPLS backbone using **labels** that are previously distributed among the core routers.
- Traffic is secure because service provider customers cannot see each other's traffic.
- There are two types of MPLS VPN solutions supported by service providers:
- **Layer 3 MPLS VPN** - The service provider participates in customer routing by establishing a **peering** between the **customer's routers** and the **provider's routers**. Then customer routes that are received by the provider's router are then redistributed through the MPLS network to the customer's remote locations.
- **Layer 2 MPLS VPN** - The service provider is not involved in the customer routing. Instead, the provider deploys a Virtual Private LAN Service (VPLS) to emulate an Ethernet multiaccess LAN segment over the MPLS network. No routing is involved. The customer's routers effectively belong to the same multiaccess network.

What is Multi-Protocol Label Switching (MPLS)?

- MPLS is a packet-forwarding technology which uses labels in order to make data forwarding decisions.
- With MPLS, the Layer 3 header analysis is done just once (when the packet enters the MPLS domain).
- Label inspection drives subsequent packet forwarding.
- MPLS provides these beneficial applications:
  - Virtual Private Networking (VPN)
  - Traffic Engineering (TE)
  - Quality of Service (QoS)
  - Any Transport over MPLS (AToM)
- Additionally, it decreases the forwarding overhead on the core routers.
- MPLS technologies are applicable to any network layer protocol.

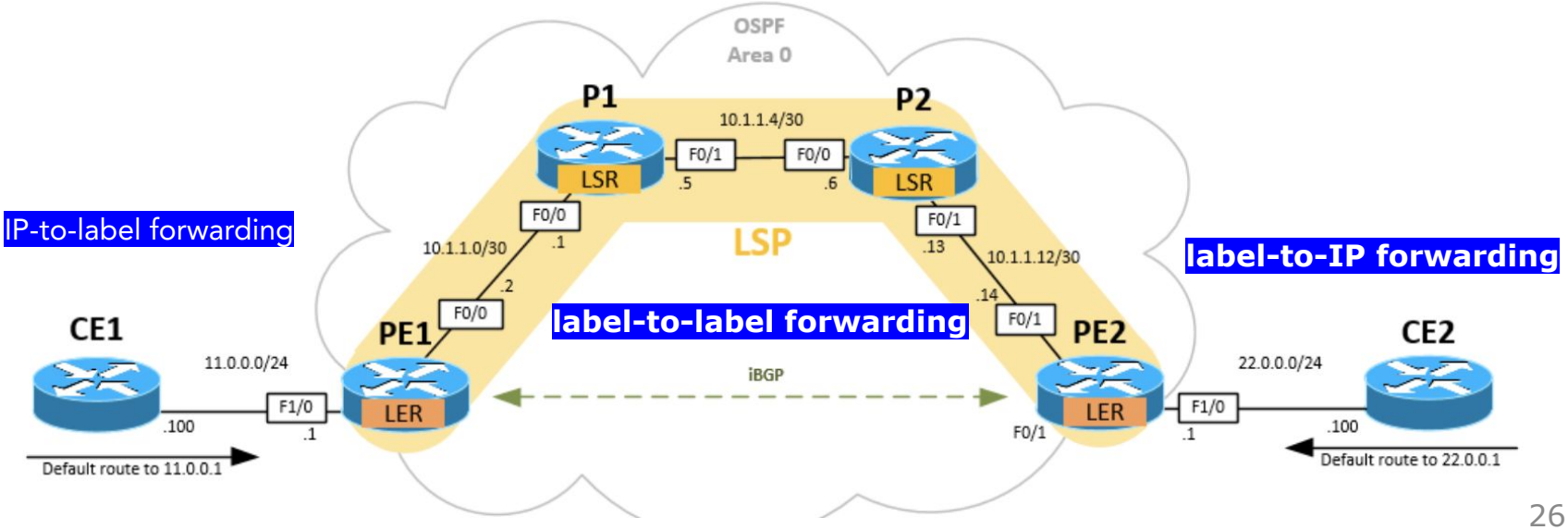
What is a label?

- A label is a short, four-byte, fixed-length, locally-significant identifier which is used in order to identify a Forwarding Equivalence Class (FEC).
- The label which is put on a particular packet represents the FEC to which that packet is assigned.
- The label is imposed between the data link layer (Layer 2) header and network layer (Layer 3) header.
- The top of the label stack appears first in the packet, and the bottom appears last.
- The network layer packet immediately follows the last label in the label stack.



How Does MPLS Work?

- In MPLS, only routers in the first and the last hop that needs to know the information about the destination prefix.
- In MPLS label is attached to the packet header as an identifier to find the way to the destination.
- The first and the last hop routers in an MPLS network are called Label Edge Router (LER).
- All the routers in between them are called Label Switching Router (LSR).
- The MPLS path established between LERs is called Label Switching Path (LSP).

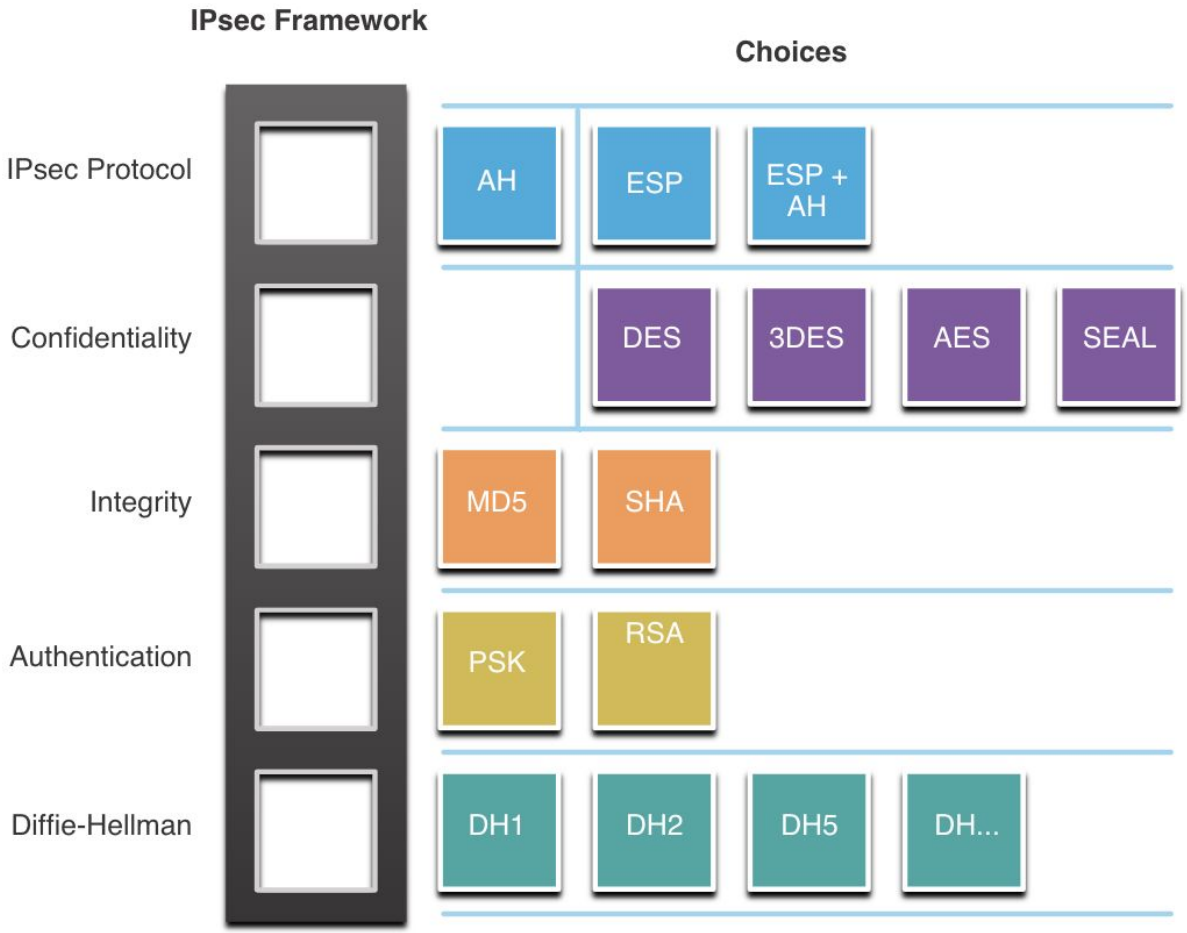


IPSec

- IPsec is an IETF standard that defines how a VPN can be secured across IP networks.
- IPsec protects and authenticates IP packets between source and destination. IPsec can protect traffic from Layer 4 through Layer 7.
- IPsec provides these essential security functions:
  - Confidentiality - IPsec uses encryption algorithms to prevent cybercriminals from reading the packet contents.
  - Integrity - IPsec uses hashing algorithms to ensure that packets have not been altered between source and destination.
  - Origin authentication - IPsec uses the Internet Key Exchange (IKE) protocol to authenticate source and destination. Methods of authentication including using pre-shared keys (passwords), digital certificates, or RSA certificates.
  - Diffie-Hellman - Secure key exchange typically using various groups of the DH algorithm.

IPsec Technologies

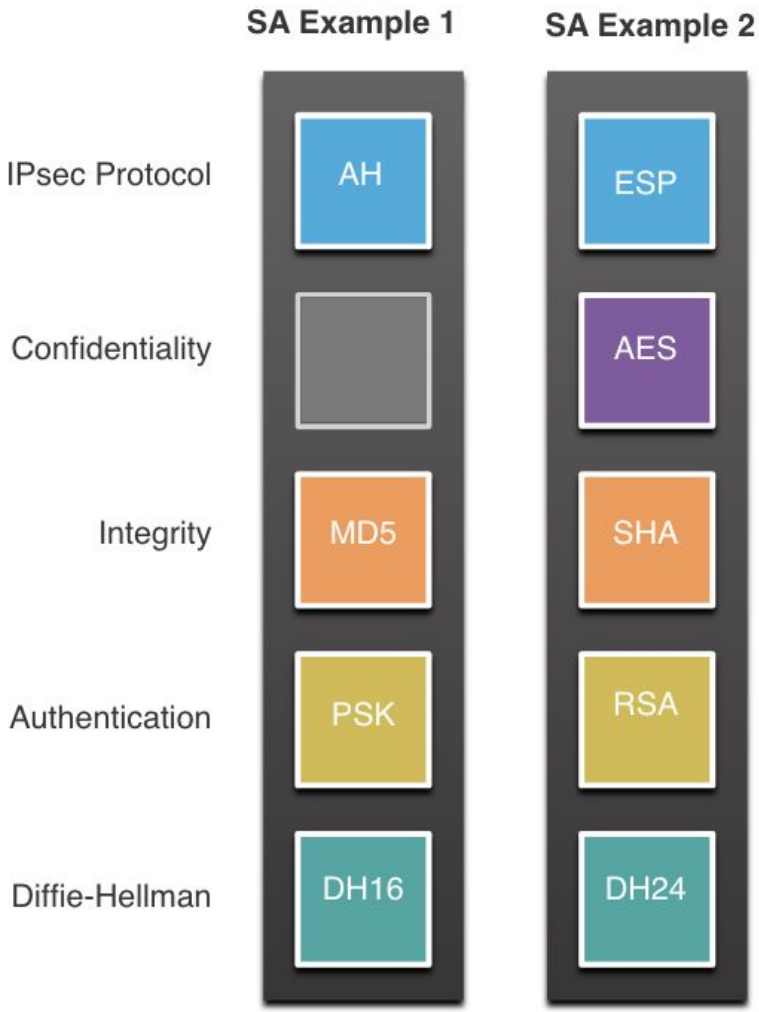
- IPsec is not bound to any specific rules for secure communications.
- This flexibility of the framework allows IPsec to easily integrate new security technologies without updating the existing IPsec standards.



IPSec Security Options

IPsec Function	Description
IPsec Protocol	The choices for IPsec Protocol include Authentication Header (AH) or Encapsulation Security Protocol (ESP). AH authenticates the Layer 3 packet. ESP encrypts the Layer 3 packet. **Note**: ESP+AH is rarely used as this combination will not successfully traverse a NAT device.
Confidentiality	Encryption ensures confidentiality of the Layer 3 packet. Choices include Data Encryption Standard (DES), Triple DES (3DES), Advanced Encryption Standard (AES), or Software-Optimized Encryption Algorithm (SEAL). No encryption is also an option.
Integrity	Ensures that data arrives unchanged at the destination using a hash algorithm, such as message-digest 5 (MD5) or Secure Hash Algorithm (SHA).
Authentication	IPsec uses Internet Key Exchange (IKE) to authenticate users and devices that can carry out communication independently. IKE uses several types of authentication, including username and password, one-time password, biometrics, pre-shared keys (PSKs), and digital certificates using the Rivest, Shamir, and Adleman (RSA) algorithm.
Diffie-Hellman	IPsec uses the DH algorithm to provide a public key exchange method for two peers to establish a shared secret key. There are several different groups to choose from including DH14, 15, 16 and DH 19, 20, 21 and 24. DH1, 2 and 5 are no longer recommended.

IPsec Security Association Examples



IPsec Protocol Encapsulation

- Choosing the IPsec protocol encapsulation is the first building block of the framework.

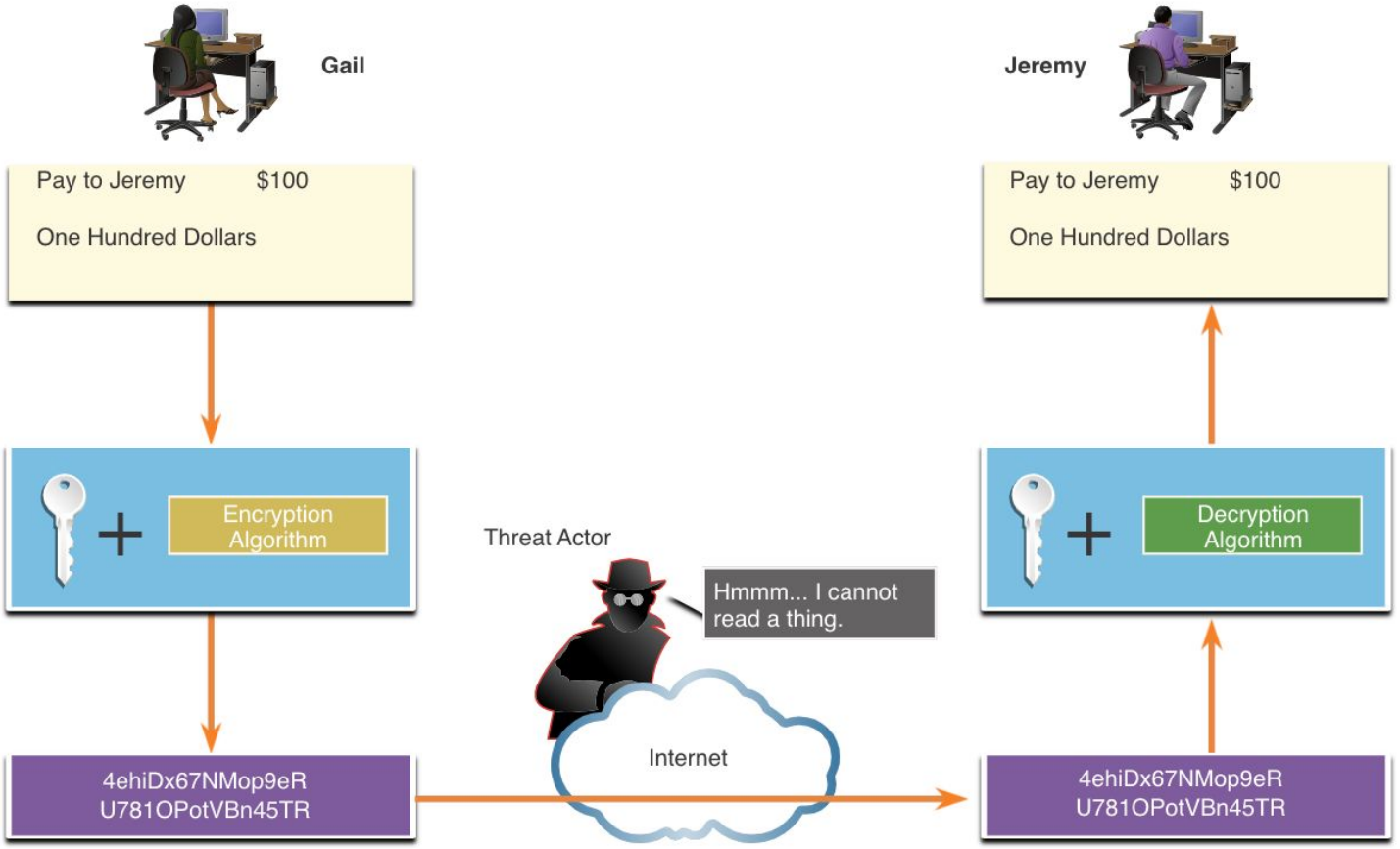
  IPsec encapsulates packets using:

  - Authentication Header (AH)
    - AH is appropriate only when confidentiality is not required or permitted.
  - Encapsulation Security Protocol (ESP)

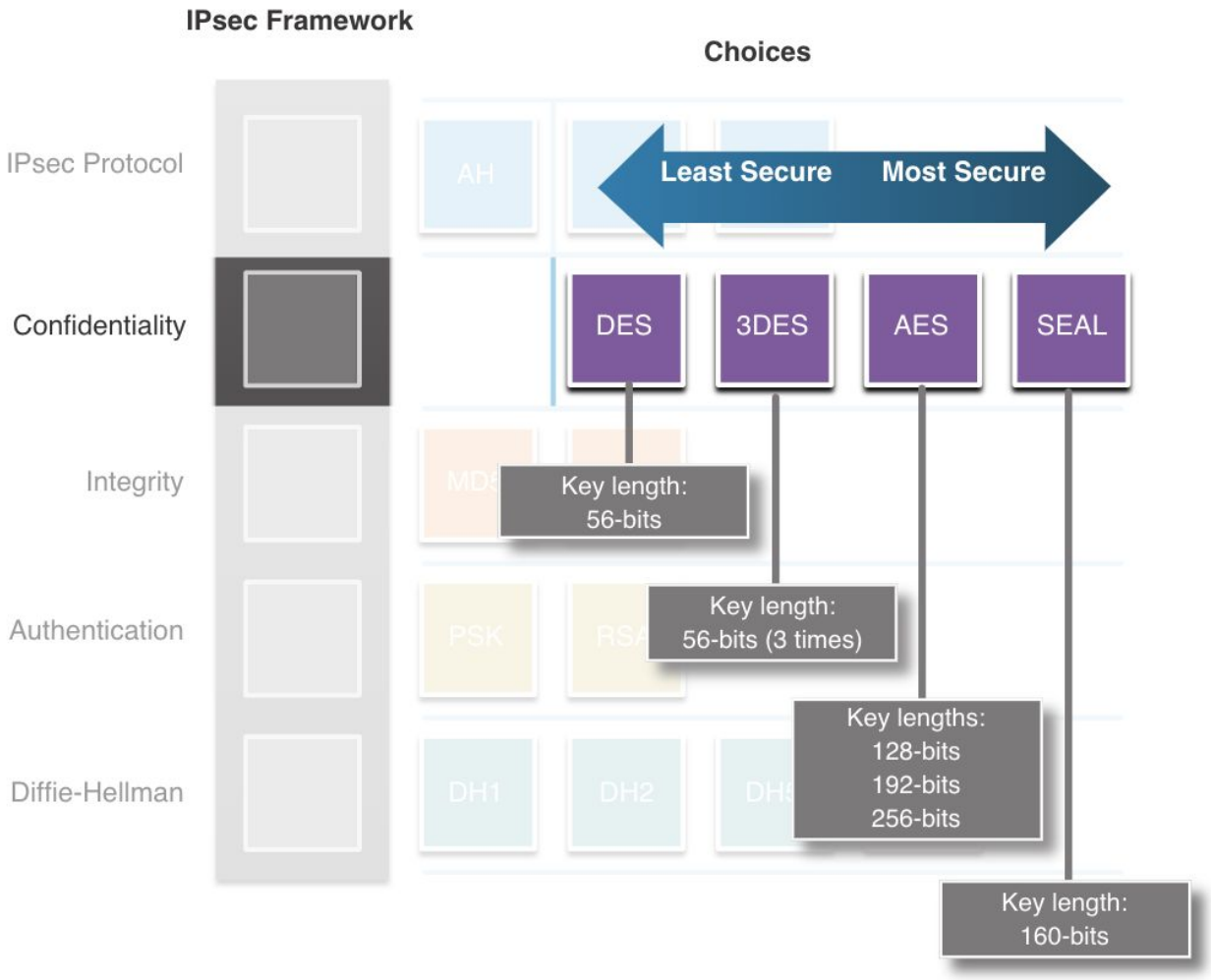
    It provides data authentication and integrity, but it does not provide data confidentiality (encryption).
- Note: All text is transported unencrypted.
- ESP provides both confidentiality and authentication.
- It provides confidentiality by performing encryption on the IP packet.
- ESP provides authentication for the inner IP packet and ESP header.
- Authentication provides data origin authentication and data integrity.
- Although both encryption and authentication are optional in ESP, at a minimum, one of them must be selected.

Confidentiality

- Confidentiality is achieved by encrypting the data.
- The degree of confidentiality depends on the encryption algorithm and the length of the key used in the encryption algorithm.

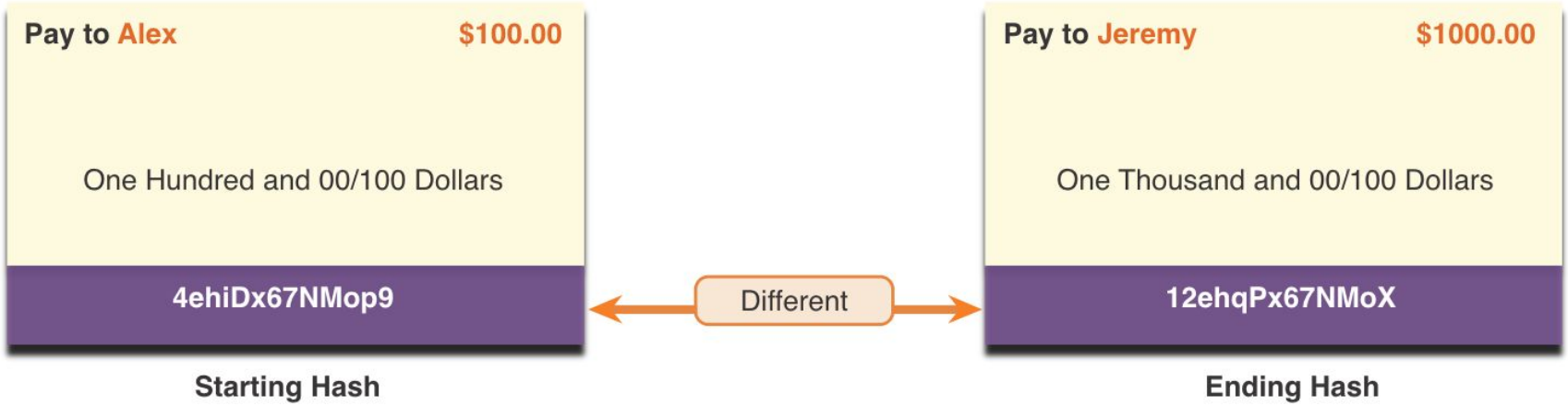


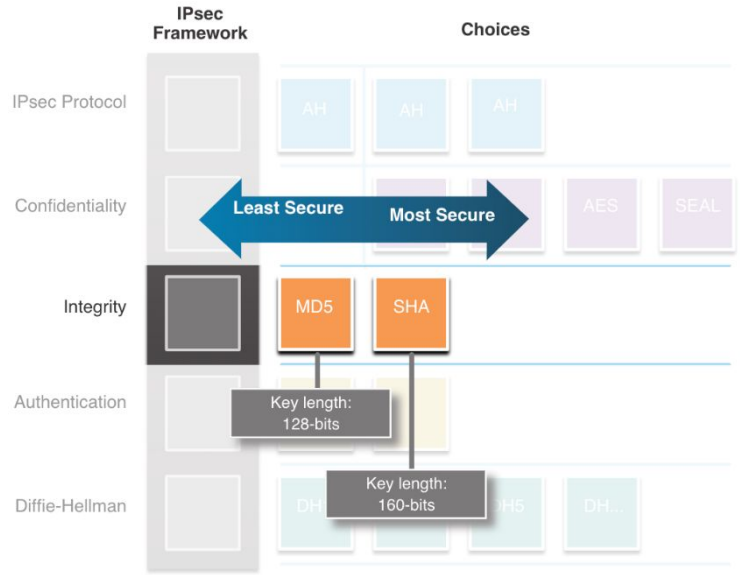
Confidentiality (cont.)



Integrity

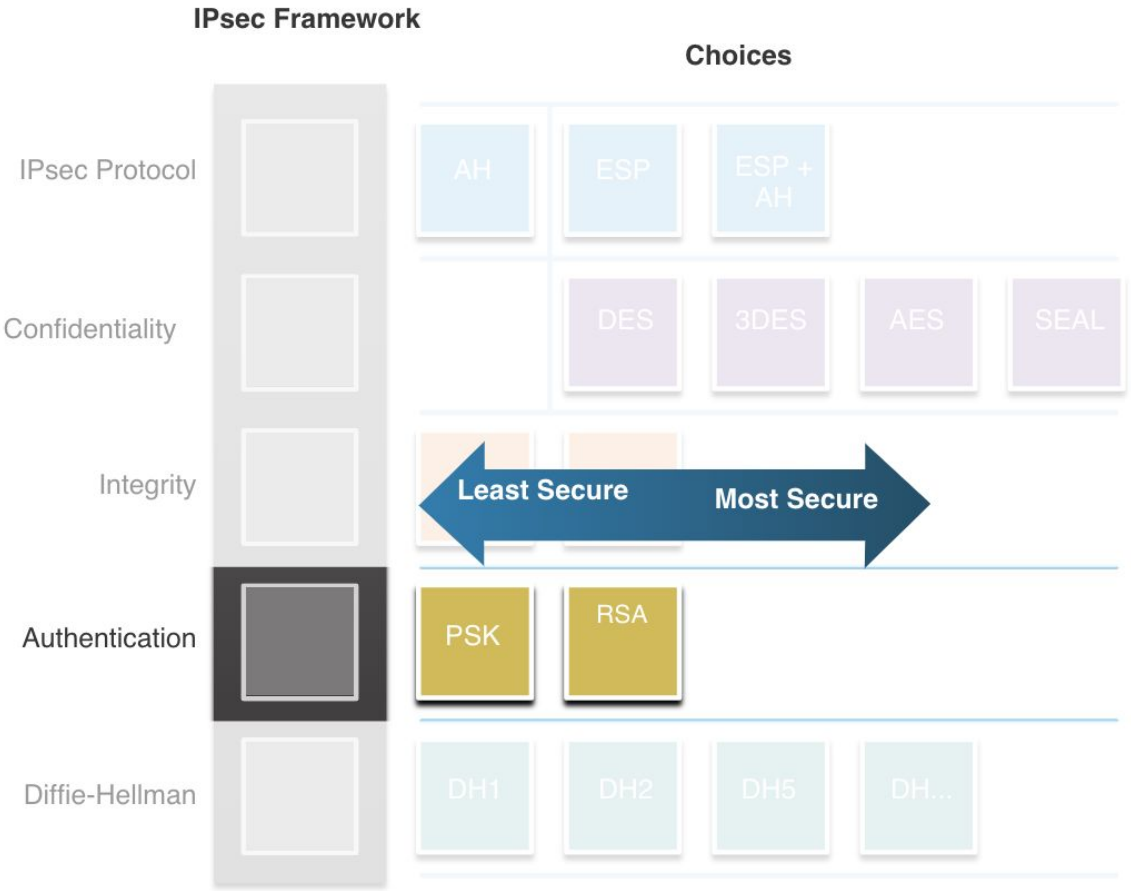
- Data integrity means that the data that is received is exactly the same data that was sent.
- Potentially, data could be intercepted and modified.





Authentication

- The device on the other end of the VPN tunnel must be authenticated before the communication path is considered secure.



IPSec VPN Configuration

Phase 1

- This is where the bidirectional ISAKMP channel is created for negotiation.
- The first thing you should create is the policy.
- A policy should contain the following:
  - Authentication method
  - Encryption algorithm
  - Hash algorithm
  - Diffie-Hellman group

We define these in a crypto ISAKMP policy like below:

< crypto isakmp policy *policy_number* authentication *auth_type* enryption *encryption_type DH_group*>

Next, we will want to specify the ISAKMP peer and the key to use to establish that ISAKMP tunnel:

crypto isakmp key *key* address *peer address*

IPSec VPN Configuration

Phase 2

In this phase we establish two unidirectional channels between the peers (IPSec SAs) so data can be sent securely.

In order for these channels to be established, the following is required:

1. Encryption algorithm
2. Hash algorithm
3. Define who the peer is
4. Apply the crypto to an interface

To define the first two requirements, you would create an IPSec transform set where you would define your encryption and hash algorithms:

cyrpto ipsec transform-set <tset-name> esp-aes 256 esp-sha512-hmac

IPSec VPN Configuration

Phase 3

Define the allowed traffic in an access-list.

This shows what traffic should be encrypted and passed through the VPN

access-list *list_nr* permit ip *source_ip destination_ip*

IPSec VPN Configuration

Phase 4

We will then tie together all of the requirements shown above in something called a crypto map which will then be applied to an interface.

crypto map <map-name> <number> ipsec-isakmp match address VPN-TRAFFIC

set peer <peer-public-ip> set transform-set <tset-name>

interface *interface_name*

Crypto map *map_name*