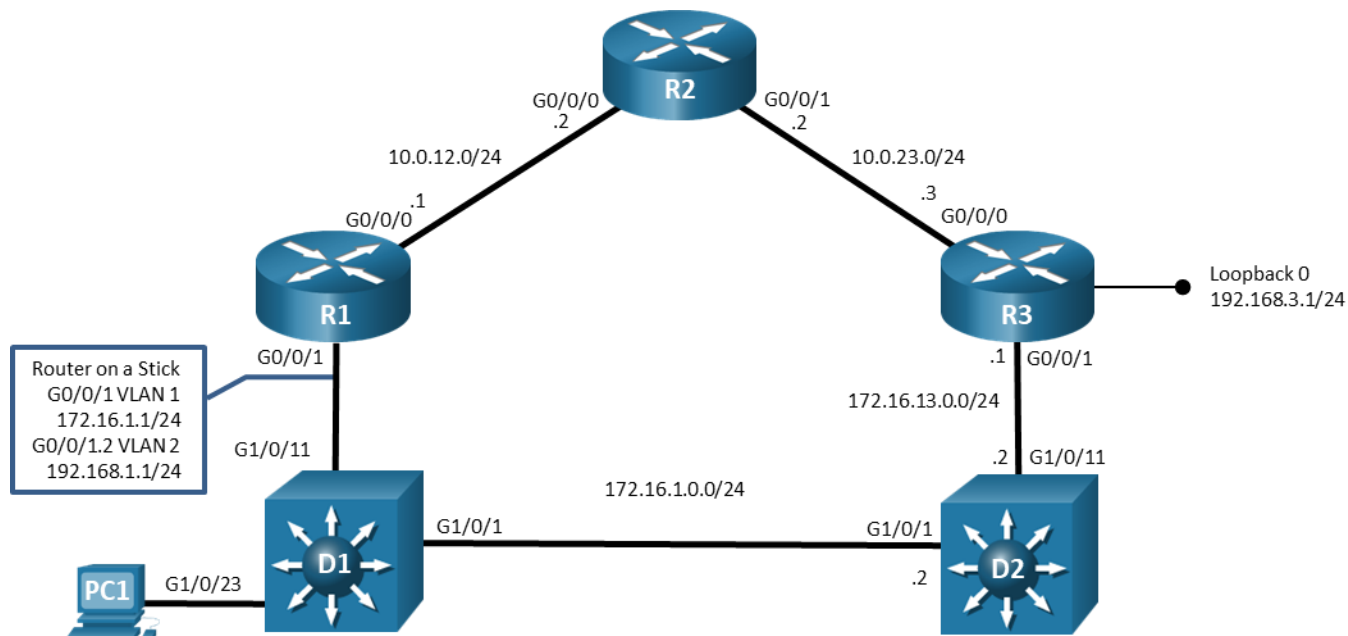


## Lab - Implement EIGRP for IPv4 (Instructor Version)

**Instructor Note:** Red font color or gray highlights indicate text that appears in the instructor copy only.

### Answers: [2.1.2 Lab - Implement EIGRP for IPv4](#)

#### Topology



#### Addressing Table

Device	Interface	IP Address	Subnet Mask
R1	G0/0/0	10.0.12.1	255.255.255.0
	G0/0/1.1	172.16.1.1	255.255.255.0
	G0/0/1.2	192.168.1.1	255.255.255.0
R2	G0/0/0	10.0.12.2	255.255.255.0
	G0/0/1	10.0.23.2	255.255.255.0
R3	G0/0/0	10.0.23.3	255.255.255.0
	G0/0/1	172.16.13.1	255.255.255.0
	Loopback 0	192.168.3.1	255.255.255.0
D2	G1/0/1	172.16.1.2	255.255.255.0
	G1/0/11	172.16.13.2	255.255.255.0
PC1	NIC	DHCP	

### Objectives

**Part 1: Build the Network and Configure Basic Device Settings**

**Part 2: Configure and Verify EIGRP for IPv4**

**Part 3: Tune EIGRP for IPv4**

### Background / Scenario

EIGRP is an interior gateway routing protocol created by Cisco and published as open source in 2016 in RFC 7868. It is a very efficient distance-vector based protocol. In this lab, you will configure and examine EIGRP in two forms, *Classic* EIGRP and *Named* EIGRP. Named EIGRP is an update to Classic EIGRP that added multiprotocol support and default support for wide metrics.

**Note:** This lab is an exercise in configuring options available for EIGRP supporting IPv4 and does not necessarily reflect implementation best practices.

**Note:** The routers used with CCNP hands-on labs are Cisco 4221 with Cisco IOS XE Release 16.9.4 (universalk9 image). The switches used in the labs are Cisco Catalyst 3650 with Cisco IOS XE Release 16.9.4 (universalk9 image). Other routers, switches, and Cisco IOS versions can be used. Depending on the model and Cisco IOS version, the commands available and the output produced might vary from what is shown in the labs. Refer to the Router Interface Summary Table at the end of the lab for the correct interface identifiers.

**Note:** Make sure that the routers and switches have been erased and have no startup configurations. If you are unsure, contact your instructor.

**Instructor Note:** Refer to the Instructor Lab Manual for the procedures to initialize and reload devices.

### Required Resources

- 3 Routers (Cisco 4221 with Cisco IOS XE Release 16.9.4 universal image or comparable)
- 2 Switches (Cisco 3650 with Cisco IOS XE Release 16.9.4 universal image or comparable)
- 1 PC (Choice of operating system with a terminal emulation program installed)
- Console cables to configure the Cisco IOS devices via the console ports
- Ethernet cables as shown in the topology

### Part 1: Build the Network and Configure Basic Device Settings

In Part 1, you will set up the network topology and configure basic settings on routers.

#### Step 1: Cable the network as shown in the topology.

Attach the devices as shown in the topology diagram, and cable as necessary.

#### Step 2: Configure basic settings for each device.

- a. Console into each device, enter global configuration mode, and apply the basic settings. The startup configurations for each device are provided below.

##### Router R1

```
hostname R1
no ip domain lookup
banner motd # R1, Implement EIGRP for IPv4 #
line con 0
  exec-timeout 0 0
```

```
logging synchronous
exit
line vty 0 4
  privilege level 15
  exec-timeout 0 0
  password cisco123
  login
  exit
interface g0/0/1
  no ip address
  no shutdown
  exit
interface g0/0/1.1
  encapsulation dot1q 1
  ip address 172.16.1.1 255.255.255.0
  no shutdown
  exit
interface g0/0/1.2
  encapsulation dot1q 2
  ip address 192.168.1.1 255.255.255.0
  no shutdown
  exit
interface g0/0/0
  ip address 10.0.12.1 255.255.255.0
  no shutdown
  exit
ip dhcp pool HOSTS
  network 192.168.1.0 255.255.255.0
  default-router 192.168.1.1
  exit
end
```

### Router R2

```
hostname R2
no ip domain lookup
banner motd # R2, Implement EIGRP for IPv4 #
line con 0
  exec-timeout 0 0
  logging synchronous
  exit
line vty 0 4
  privilege level 15
  exec-timeout 0 0
  password cisco123
  login
  exit
```

```
interface g0/0/0
  ip address 10.0.12.2 255.255.255.0
  no shutdown
  exit
interface g0/0/1
  ip address 10.0.23.2 255.255.255.0
  no shutdown
  exit
end
```

### Router R3

```
hostname R3
no ip domain lookup
banner motd # R3, Implement EIGRP for IPv4 #
line con 0
  exec-timeout 0 0
  logging synchronous
  exit
line vty 0 4
  privilege level 15
  exec-timeout 0 0
  password cisco123
  login
  exit
interface g0/0/0
  ip address 10.0.23.3 255.255.255.0
  no shutdown
  exit
interface g0/0/1
  ip address 172.16.13.1 255.255.255.0
  no shutdown
  exit
interface loopback 0
  ip address 192.168.3.1 255.255.255.0
  no shutdown
  exit
end
```

### Switch D1

```
hostname D1
no ip domain lookup
banner motd # D1, Implement EIGRP for IPv4 #
line con 0
  exec-timeout 0 0
  logging synchronous
  exit
```

```
line vty 0 4
  privilege level 15
  exec-timeout 0 0
  password cisco123
  login
  exit
vlan 2
  name HOST-VLAN
  exit
interface range g1/0/1 - 24, g1/1/1 - 4
  shutdown
  exit
interface g1/0/1
  switchport mode access
  spanning-tree portfast
  no shutdown
  exit
interface g1/0/11
  switchport mode trunk
  no shutdown
  exit
interface g1/0/23
  switchport mode access
  switchport access vlan 2
  spanning-tree portfast
  no shutdown
  exit
end
```

### Switch D2

```
hostname D2
no ip domain lookup
ip routing
banner motd # D2, Implement EIGRP for IPv4 #
line con 0
  exec-timeout 0 0
  logging synchronous
  exit
line vty 0 4
  privilege level 15
  exec-timeout 0 0
  password cisco123
  login
  exit
interface range g1/0/1 - 24, g1/1/1 - 4
  shutdown
```

```
exit
interface g1/0/1
no switchport
ip address 172.16.1.2 255.255.255.0
no shutdown
exit
interface g1/0/11
no switchport
ip address 172.16.13.2 255.255.255.0
no shutdown
exit
end
```

- b. Set the clock on each device to UTC time.
- c. Save the running configuration to startup-config.
- d. Verify that PC1 receives an address via DHCP.
- e. Verify that PC1 can ping its default gateway.

### Part 2: Configure and Verify EIGRP for IPv4

In this part of the lab, you will configure and verify EIGRP in the network. R1 and R3 will use Named EIGRP, while R2 will use Classic EIGRP. After you have established the network, you will examine the differences in how each version of EIGRP deals with metrics.

For the lab, you will use the Autonomous System number 27 on all routers.

#### Step 1: Configure Classic EIGRP for IPv4 on R2.

- a. Start the configuration of Classic EIGRP by issuing the **router eigrp 27** command.

```
R2(config)# router eigrp 27
```

- b. Configure the EIGRP router ID using the **eigrp router-id** command. Use the number 2.2.2.2 for R2.

```
R2(config-router)# eigrp router-id 2.2.2.2
```

- c. Identify the interfaces that should be speaking EIGRP and the networks that should be included in the EIGRP topology table. This is done with the **network** command.

It is best to be as specific as possible when creating network statements, while balancing efficiency and the number of commands necessary. For our lab example, we will use **network 10.0.0.0 255.255.224.0** to specify the interfaces. This covers less network space than 10.0.0.0/8, while including both interfaces with a single **network** command.

```
R2(config-router)# network 10.0.0.0 255.255.224.0
```

```
R2(config-router)# end
```

- d. Verify the interfaces now involved in EIGRP with the **show ip eigrp interfaces** command.

```
R2# show ip eigrp interfaces
```

```
EIGRP-IPv4 Interfaces for AS(27)
```

		Xmit Queue	PeerQ	Mean	Pacing Time	Multicast
Pending						
Interface	Peers	Un/Reliable	Un/Reliable	SRTT	Un/Reliable	Flow
Timer Routes						
Gi0/0/0	0	0/0	0/0	0	0/0 0	0

```
Gi0/0/1          0          0/0          0/0          0          0/0 0          0
```

### Step 2: Configure Named EIGRP for IPv4 on R1 and R3.

- Start the configuration of Named EIGRP by issuing the **router eigrp [ name ]** command. The name parameter can be a number, but the number does not identify an Autonomous System as it does with Classic EIGRP, it simply identifies the process. For our purposes, name the process BASIC-EIGRP-LAB.

```
R1(config)# router eigrp BASIC-EIGRP-LAB
```

- Enter into address-family configuration mode with the **address-family ipv4 unicast autonomous-system 27** command.

```
R1(config-router)# address-family ipv4 unicast autonomous-system 27
```

- Configure the EIGRP router ID using the **eigrp router-id** command. Use the number 1.1.1.1 for R1.

```
R1(config-router-af)# eigrp router-id 1.1.1.1
```

- Identify the interfaces that should be speaking EIGRP and the networks that should be included in the EIGRP topology table. This is done with the **network** command. In this case, the configuration requires three **network** commands. An example for R1 follows:

```
R1(config-router-af)# network 10.0.12.0 255.255.255.0
```

```
R1(config-router-af)# network 172.16.1.0 255.255.255.0
```

```
R1(config-router-af)# network 192.168.1.0 255.255.255.0
```

```
R1(config-router-af)# end
```

- Repeat Steps 2a through 2d for R3 and D2. Use 3.3.3.3 for the router ID on R3, and 132.132.132.132 for the router ID on D2. Configure the appropriate network statements on both devices according to the Addressing Table.

```
R3(config)# router eigrp BASIC-EIGRP-LAB
```

```
R3(config-router)# address-family ipv4 unicast autonomous-system 27
```

```
R3(config-router-af)# eigrp router-id 3.3.3.3
```

```
R3(config-router-af)# network 10.0.23.0 255.255.255.0
```

```
R3(config-router-af)# network 172.16.13.0 255.255.255.0
```

```
R3(config-router-af)# network 192.168.3.0 255.255.255.0
```

```
R3(config-router-af)# end
```

```
D2(config-if)# router eigrp BASIC-EIGRP-LAB
```

```
D2(config-router)# address-family ipv4 unicast autonomous-system 27
```

```
D2(config-router-af)# eigrp router-id 132.132.132.132
```

```
D2(config-router-af)# network 172.16.0.0 255.255.0.0
```

```
D2(config-router-af)# end
```

### Step 3: Verify EIGRP for IPv4.

- A few seconds after configuring the network statements, you should have seen syslog messages noting that EIGRP adjacencies have been formed.

```
*Feb 18 15:49:34.243: %DUAL-5-NBRCHANGE: EIGRP-IPv4 27: Neighbor 10.0.12.2  
(GigabitEthernet0/0/0) is up: new adjacency
```

- To verify that routing is working, ping from PC1 to interface Loopback 0 on R3 (192.168.3.1). The ping should be successful. You can also randomly ping other addresses in the topology.
- On R1, examine the EIGRP entries in the IP routing table using the **show ip route eigrp | begin Gateway** command. As you can see, there is one path installed in the routing table for the network, and two paths for the 192.168.3.0/24 network. Take note of the metric values listed.

```
R1# show ip route eigrp | begin Gateway
```

```
Gateway of last resort is not set
```

```
10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
D    10.0.23.0/24 [90/15360] via 10.0.12.2, 00:00:12, GigabitEthernet0/0/0
172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
D    172.16.13.0/24
    [90/15360] via 172.16.1.2, 00:00:12, GigabitEthernet0/0/1.1
D    192.168.3.0/24
    [90/16000] via 172.16.1.2, 00:00:12, GigabitEthernet0/0/1.1
    [90/16000] via 10.0.12.2, 00:00:12, GigabitEthernet0/0/0
```

- d. Now examine the EIGRP topology table using the **show ip eigrp topology all-links** command. The **all-links** parameter instructs the router to display all available routes, including the ones that are not successors or feasible successors.

Remember that the topology table is EIGRP's database of route information. EIGRP selects the best paths from this database, based on the DUAL algorithm, and offers them to the IP routing table. However, the IP routing table does not have to use those offered paths, because the router may have learned about the same network from a more reliable routing source, which would be a routing source with a lower administrative distance.

```
R1# show ip eigrp topology all-links
```

```
EIGRP-IPv4 VR(BASIC-EIGRP-LAB) Topology Table for AS(27)/ID(1.1.1.1)
```

```
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
```

```
P 192.168.3.0/24, 2 successors, FD is 2048000, serno 26
   via 10.0.12.2 (2048000/1392640), GigabitEthernet0/0/0
   via 172.16.1.2 (2048000/1392640), GigabitEthernet0/0/1.1
P 172.16.13.0/24, 1 successors, FD is 1966080, serno 16
   via 172.16.1.2 (1966080/1310720), GigabitEthernet0/0/1.1
   via 10.0.12.2 (2621440/1966080), GigabitEthernet0/0/0
P 192.168.1.0/24, 1 successors, FD is 1310720, serno 3
   via Connected, GigabitEthernet0/0/1.2
P 172.16.1.0/24, 1 successors, FD is 1310720, serno 2
   via Connected, GigabitEthernet0/0/1.1
P 10.0.23.0/24, 1 successors, FD is 1966080, serno 27
   via 10.0.12.2 (1966080/1310720), GigabitEthernet0/0/0
   via 172.16.1.2 (2621440/1966080), GigabitEthernet0/0/1.1
P 10.0.12.0/24, 1 successors, FD is 1310720, serno 1
   via Connected, GigabitEthernet0/0/0
```

We will focus on the routes, highlighted in the above output, to 192.168.3.0/24 and 10.0.23.0/24. There are several things to notice:

- The entry for the 192.168.3.0/24.0 network shows two successors, while the entry for 10.0.23.0/24 shows only one successor. Both entries show two paths. The path with the lowest Feasible Distance (FD) is selected as the successor and is offered to the routing table. For 192.168.3.0/24, there are two paths with equal FD (2048000). Therefore, both are successors and both are offered to the routing table. In the case of 10.0.23.0/24, the FD is listed as 1966080. The path via 10.0.12.2 shows that number as the FD (first number in parentheses). The path via 172.16.1.2 shows an FD of 2621440, which is higher than the current feasible distance. So that path, although valid, is a higher cost path and is not offered to the routing table.



- The FD listed in the topology table does not match the metric listed in the routing table. For 192.168.3.0/24, the routing table shows the metric value 16000, while the topology table shows the FD as 2048000. This is due to the routing table having a limit of 4 bytes (32 bits) for metric information while EIGRP on R1 is using EIGRP wide metrics, which are 64 bits. Wide metrics are used by Named EIGRP by default. To work around the 32-bit metric size limitation in the routing table, EIGRP divides the wide-metric value by the EIGRP\_RIB\_SCALE value, which defaults to 128, as you will see next in the output for the **show ip protocols** command. The value 2048000 divided by 128 is 16000.

**Note:** A network with mixed EIGRP implementations (Named and Classic in the same routing domain), will have some loss of route clarity which could lead to sub-optimal path selection. The recommended implementation is to use Named EIGRP in all cases.

- There are no feasible successors listed in the topology table for 192.168.3.0/24 or 10.0.23.0/24. The feasibility condition requires that the reported distance (RD) to a destination network be less than the current feasible distance for a next-hop to be considered a feasible successor to the route. In the case of 10.0.23.0/24, the RD of the path via 172.16.1.2 is the second number listed: (2621440/1966080). Because the RD is equal to the current FD, this route is disqualified as a feasible successor. If the path via 172.16.1.2 were to be lost, R1 would have to send queries to find a new way to get to 10.0.23.0/24. Feasible successors only exist in the topology table. Only successors appear in the routing table.
- e. To see the Routing Information Base (RIB) Scale and Metric Scale values, as well as other protocol information, issue the **show ip protocols | section eigrp** command.

```
R1# show ip protocols | section eigrp
```

```
Routing Protocol is "eigrp 27"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP-IPv4 VR(BASIC-EIGRP-LAB) Address-Family Protocol for AS(27)
    Metric weight K1=1, K2=0, K3=1, K4=0, K5=0 K6=0
    Metric rib-scale 128
    Metric version 64bit
    Soft SIA disabled
    NSF-aware route hold timer is 240
  EIGRP NSF disabled
    NSF signal timer is 20s
    NSF converge timer is 120s
  Router-ID: 1.1.1.1
  Topology : 0 (base)
    Active Timer: 3 min
    Distance: internal 90 external 170
    Maximum path: 4
    Maximum hopcount 100
    Maximum metric variance 1
    Total Prefix Count: 5
    Total Redist Count: 0
```

- f. To examine details about a particular path, issue the **show ip eigrp topology [address]** command.

```
R1# show ip eigrp topology 192.168.3.0/24
```

```
EIGRP-IPv4 VR(BASIC-EIGRP-LAB) Topology Entry for AS(27)/ID(1.1.1.1) for
192.168.3.0/24
```

## Lab - Implement EIGRP for IPv4

---

```
State is Passive, Query origin flag is 1, 2 Successor(s), FD is 2048000, RIB is
16000
Descriptor Blocks:
10.0.12.2 (GigabitEthernet0/0/0), from 10.0.12.2, Send flag is 0x0
  Composite metric is (2048000/1392640), route is Internal
  Vector metric:
    Minimum bandwidth is 1000000 Kbit
    Total delay is 21250000 picoseconds
    Reliability is 255/255
    Load is 1/255
    Minimum MTU is 1500
    Hop count is 2
    Originating router is 3.3.3.3
172.16.1.2 (GigabitEthernet0/0/1.1), from 172.16.1.2, Send flag is 0x0
  Composite metric is (2048000/1392640), route is Internal
  Vector metric:
    Minimum bandwidth is 1000000 Kbit
    Total delay is 21250000 picoseconds
    Reliability is 255/255
    Load is 1/255
    Minimum MTU is 1500
    Hop count is 2
    Originating router is 3.3.3.3
```

### Part 3: Tune EIGRP for IPv4

In this part of the lab, you will tune and optimize EIGRP for IPv4 through the use of passive interfaces, authentication, and variance.

#### Step 1: Configure specific interfaces as passive.

Passive interfaces are interfaces that only partially participate in a routing protocols operation. The network address for the passive interface is advertised through other interfaces. However, the routing protocol does not actually speak on the passive interface. Use passive interfaces when you have a connected network that you want to advertise, but you do not want protocol neighbors to appear on that interface. For example, interfaces supporting users should always be configured as passive. There are two ways to configure interfaces as passive, either specifically by interface name or by default. Normally a device with many LAN interfaces will use the default option, and then use the **no** form of the command on those specific interfaces that should be sending and receiving EIGRP messages.

- a. On PC1, run Wireshark and set the capture filter to **eigrp**. You should see a hello message roughly every five seconds. If your device were capable of running EIGRP, you might be able to form an adjacency and interact in the routing domain. This is unnecessary traffic on the LAN and a potential security risk. Stop the capture.
- b. On R1, configure af-interface G0/0/1 to be passive.

```
R1(config)# router eigrp BASIC-EIGRP-LAB
R1(config-router)# address-family ipv4 unicast autonomous-system 27
R1(config-router-af)# af-interface g0/0/1.2
R1(config-router-af-interface)# passive-interface
R1(config-router-af-interface)# end
```

- c. On PC1, restart the Wireshark capture with the capture filter still configured for **eigrp**. You should no longer see EIGRP Hello messages.

### Step 2: Configure interfaces to default to passive.

The second option for configuring passive interfaces is to configure them all as passive and then issue the `no passive-interface` command for certain interfaces. This approach is suitable in a security-focused scenario, or when the device has many LAN interfaces. The commands vary depending on whether you are using Classic or Named EIGRP.

- a. In Classic EIGRP configuration, enter the **passive-interface default** command followed by **no passive-interface *interface-number*** commands for all the interfaces that should be participating in EIGRP. The following provides an example of this on R2. You will temporarily lose EIGRP adjacencies until **passive-interface** is removed from required interfaces.

```
R2(config)# router eigrp 27
R2(config-router)# passive-interface default
R2(config-router)# no passive-interface g0/0/0
R2(config-router)# no passive-interface g0/0/1
R2(config-router)# exit
R2(config)# end
```

- b. In Named EIGRP configuration, you apply the **passive-interface** command to the **af-interface default** configuration, and the **no passive-interface** command to the specific **af-interface *interface-number***. The following provides an example of this on R3. You will temporarily lose EIGRP adjacencies until **passive-interface** is removed from required interfaces.

```
R3(config)# router eigrp BASIC-EIGRP-LAB
R3(config-router)# address-family ipv4 unicast autonomous-system 27
R3(config-router-af)# af-interface default
R3(config-router-af-interface)# passive-interface
R3(config-router-af-interface)# exit
R3(config-router-af)# af-interface g0/0/0
R3(config-router-af-interface)# no passive-interface
R3(config-router-af-interface)# exit
R3(config-router-af)# af-interface g0/0/1
R3(config-router-af-interface)# no passive-interface
R3(config-router-af-interface)# end
```

- c. The output of the **show ip protocols | section Passive** command will give you a list of passive interfaces configured for EIGRP, which for R3 will only be the Loopback 0 interface that is simulating a LAN.

```
R3# show ip protocols | section Passive
  Passive Interface(s):
    Loopback0
```

### Step 3: Configure EIGRP authentication.

EIGRP supports authentication on an interface basis. In other words, each interface can be configured to require authentication of the connected peer. This ensures that connected devices that try to form an adjacency are authorized to do so. Classic EIGRP supports key-chain based MD5-hashed keys, while Named EIGRP adds support for SHA256-hashed keys. The two are not compatible.

In this step, you will configure both types of authentication to exercise the range of options available.

- a. On R1, R2, R3, and D2, create a key-chain named EIGRP-AUTHEN-KEY with a single key. The key should have the key-string **\$3cre7!!**

```
R1(config)# key chain EIGRP-AUTHEN-KEY
R1(config-keychain)# key 1
R1(config-keychain-key)# key-string $3cre7!!
R1(config-keychain-key)# end
```

- b. On R2, configure interfaces G0/0/0 and G0/0/1 to encrypt the key chain you just created with MD5. Note that you will lose EIGRP adjacencies until the neighbor interfaces are configured.

```
R2(config)# interface g0/0/0
R2(config-if)# ip authentication key-chain eigrp 27 EIGRP-AUTHEN-KEY
R2(config-if)# ip authentication mode eigrp 27 md5
R2(config-if)# exit
R2(config)# interface g0/0/1
R2(config-if)# ip authentication key-chain eigrp 27 EIGRP-AUTHEN-KEY
R2(config-if)# ip authentication mode eigrp 27 md5
R2(config-if)# end
```

- c. Configure interface G0/0/0 on both R1 and R3 to use the key chain with MD5. EIGRP adjacencies with R2 should be restored.

```
R1(config)# router eigrp BASIC-EIGRP-LAB
R1(config-router)# address-family ipv4 unicast autonomous-system 27
R1(config-router-af)# af-interface g0/0/0
R1(config-router-af-interface)# authentication key-chain EIGRP-AUTHEN-KEY
R1(config-router-af-interface)# authentication mode md5
R1(config-router-af-interface)# end
```

- d. Use the **show ip eigrp interface detail | section Gi0/0/0** command to verify that authentication is in place and what type of authentication it is.

```
R1# show ip eigrp interface detail | section Gi0/0/0
Gi0/0/0          1          0/0          0/0          1          0/050          0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 14/2
  Hello's sent/expedited: 186/4
  Un/reliable mcasts: 0/11  Un/reliable ucasts: 15/7
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 3  Out-of-sequence rcvd: 0
  Topology-ids on interface - 0
  Authentication mode is md5, key-chain is "EIGRP-AUTHEN-KEY"
  Topologies advertised on this interface:  base
  Topologies not advertised on this interface:
```

- e. On R1 and D2, configure SHA 256 based authentication using the same **\$3cre7!!** shared secret.. R1 and D2 are running Named EIGRP, so the configuration is applied in **af-interface** mode. On R1, configure the G0/0/1.1 subinterface. On D2, configure the G1/0/1 interface. The configuration for R1 is shown. Note that the R1-D2 adjacency will be lost until both ends are configured.

```
R1(config)# router eigrp BASIC-EIGRP-LAB
```

```
R1(config-router)# address-family ipv4 unicast autonomous-system 27
R1(config-router-af)# af-interface g0/0/1.1
R1(config-router-af-interface)# authentication mode hmac-sha-256 $3cre7!!
R1(config-router-af-interface)# end
```

```
D2(config)# router eigrp BASIC-EIGRP-LAB
D2(config-router)# address-family ipv4 unicast autonomous-system 27
D2(config-router-af)# af-interface g1/0/1
D2(config-router-af-interface)# authentication mode hmac-sha-256 $3cre7!!
D2(config-router-af-interface)# end
```

- f. Use the **show ip eigrp interface detail** command to verify that authentication is in place and what type of authentication it is.

```
R1# show ip eigrp interface detail | section Gi0/0/1.1
Gi0/0/1.1          1          0/0          0/0          3          0/050          0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 11/0
  Hello's sent/expedited: 225/4
  Un/reliable mcasts: 0/11  Un/reliable ucasts: 12/7
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 4  Out-of-sequence rcvd: 0
  Topology-ids on interface - 0
  Authentication mode is HMAC-SHA-256, key-chain is not set
  Topologies advertised on this interface:  base
  Topologies not advertised on this interface:
```

### Step 4: Manipulate load balancing with variance

By default, load balancing occurs only over equal-cost paths. EIGRP supports up to four equal cost paths by default but can be configured to support as many as 32 with the **maximum-paths** command.

EIGRP has the added capability to load balance over unequal-cost paths. Load balancing is controlled by the **variance** parameter. Its value is a multiplier that is used to determine how to deal with multiple paths to the same destination.

Variance is set to 1 by default, so any paths up to the configured maximum number of paths that have a feasible distance equal to the best current feasible distance are also offered to the routing table. This provides equal cost load balancing.

The variance parameter can also be set to zero, which dictates that no load balancing takes place.

The variance parameter can be adjusted so that paths that have an FD that is less than or equal to variance times current best FD are also considered as successors and installed into the routing table. There is an extremely important differentiation here -- to be a feasible successor, the RD of a path must be less than the current best FD. To be considered for unequal load balancing, the FD of the feasible successor is multiplied by the variance value, and if the product of this calculation is less than the current best FD, the feasible successor is promoted to successor.

There are two caveats; first, only feasible successors are considered and second, unequal cost load balancing is unequal; traffic share is proportional to the best metric in the routing table for the given path.

**Note:** Keep in mind that your routing table may be different than the one created by the examples in this lab. If your results are different, examine them carefully to determine why so that you can thoroughly understand how EIGRP is operating.

- a. On R3, there are two equal-cost paths to 192.168.1.0/24.

```
R3# show ip route eigrp | section 192.168.1.0
D    192.168.1.0/24
      [90/20480] via 172.16.13.2, 00:08:18, GigabitEthernet0/0/1
      [90/20480] via 10.0.23.2, 00:08:18, GigabitEthernet0/0/0
```

- b. To change this and allow for the demonstration of variance, change the interface bandwidth for the R2 interfaces G0/0/0 and G0/0/1 to 800000.

```
R2(config)# interface g0/0/0
R2(config-if)# bandwidth 800000
R2(config-if)# exit
R2(config)# interface g0/0/1
R2(config-if)# bandwidth 800000
R2(config-if)# end
```

- c. When you examine the routing table on R3, you see that there is no load balancing occurring. All destinations have a single path.

```
R3# show ip route eigrp | begin Gateway
Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
D    10.0.12.0/24 [90/16640] via 10.0.23.2, 00:01:17, GigabitEthernet0/0/0
    172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
D    172.16.1.0/24
      [90/15360] via 172.16.13.2, 00:01:04, GigabitEthernet0/0/1
D    192.168.1.0/24
      [90/20480] via 172.16.13.2, 00:01:04, GigabitEthernet0/0/1
```

- d. However, we know there are multiple paths in the network. The first consideration for manipulating variance is that it only works with feasible successors. Examining the topology table on R3 shows that there is a feasible successor for the 192.168.1.0/24 network. The route via 10.0.23.2 out the G0/0/0 interface has a RD less than the FD for the current successor.

```
R3# show ip eigrp topology | section 192.168.1.0
P 192.168.1.0/24, 1 successors, FD is 2621440
   via 172.16.13.2 (2621440/1966080), GigabitEthernet0/0/1
   via 10.0.23.2 (2785280/2129920), GigabitEthernet0/0/0
```

- e. To use the other route for unequal cost load balancing, we can set the variance parameter to 2. This will mean that any path with an RD less than or equal to 5242880 will qualify as a successor ( $2 \times 2621440 = 5242880$ ).

```
R3(config)# router eigrp BASIC-EIGRP-LAB
R3(config-router)# address-family ipv4 unicast autonomous-system 27
R3(config-router-af)# topology base
R3(config-router-af-topology)# variance 2
R3(config-router-af-topology)# end
```

## Lab - Implement EIGRP for IPv4

- f. The output of the **show ip route eigrp | begin Gateway** command now displays two paths available to the 192.168.1.0/24 network. Notice that the routes have different metrics, but are listed and used just the same. Also, notice adding **variance 2** adds a second path to the 10.0.12.0/24 network.

```
R3# show ip route eigrp | begin Gateway
Gateway of last resort is not set

 10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
D    10.0.12.0/24
      [90/20480] via 172.16.13.2, 00:00:11, GigabitEthernet0/0/1
      [90/16640] via 10.0.23.2, 00:00:11, GigabitEthernet0/0/0
 172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
D    172.16.1.0/24
      [90/15360] via 172.16.13.2, 00:00:11, GigabitEthernet0/0/1
D    192.168.1.0/24
      [90/20480] via 172.16.13.2, 00:00:11, GigabitEthernet0/0/1
      [90/21760] via 10.0.23.2, 00:00:11, GigabitEthernet0/0/0
```

- g. Issue the **show ip route 192.168.1.0** command to see more details about the paths the router has to the 192.168.1.0 network. As a part of this output, you see the traffic share count, which tells you the ratio of traffic that will be sent between these links. In this example, the count is 120 via 172.16.13.2 and 113 via 10.0.23.2. What that means is that 120 packets will be sent via 172.16.13.2 and then 113 packets will be sent via 10.0.23.2.

```
R3# show ip route 192.168.1.0
Routing entry for 192.168.1.0/24
  Known via "eigrp 27", distance 90, metric 20480, type internal
  Redistributing via eigrp 27
  Last update from 10.0.23.2 on GigabitEthernet0/0/0, 00:01:42 ago
  Routing Descriptor Blocks:
  * 172.16.13.2, from 172.16.13.2, 00:01:42 ago, via GigabitEthernet0/0/1
    Route metric is 20480, traffic share count is 120
    Total delay is 30 microseconds, minimum bandwidth is 1000000 Kbit
    Reliability 255/255, minimum MTU 1500 bytes
    Loading 1/255, Hops 2
  10.0.23.2, from 10.0.23.2, 00:01:42 ago, via GigabitEthernet0/0/0
    Route metric is 21760, traffic share count is 113
    Total delay is 30 microseconds, minimum bandwidth is 800000 Kbit
    Reliability 255/255, minimum MTU 1500 bytes
    Loading 1/255, Hops 2
```

### Router Interface Summary Table

Router Model	Ethernet Interface #1	Ethernet Interface #2	Serial Interface #1	Serial Interface #2
1800	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
1900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)

Router Model	Ethernet Interface #1	Ethernet Interface #2	Serial Interface #1	Serial Interface #2
2801	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)
2811	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
4221	Gigabit Ethernet 0/0/0 (G0/0/0)	Gigabit Ethernet 0/0/1 (G0/0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)
4300	Gigabit Ethernet 0/0/0 (G0/0/0)	Gigabit Ethernet 0/0/1 (G0/0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)

**Note:** To find out how the router is configured, look at the interfaces to identify the type of router and how many interfaces the router has. There is no way to effectively list all the combinations of configurations for each router class. This table includes identifiers for the possible combinations of Ethernet and Serial interfaces in the device. The table does not include any other type of interface, even though a specific router may contain one. An example of this might be an ISDN BRI interface. The string in parenthesis is the legal abbreviation that can be used in Cisco IOS commands to represent the interface.

## Device Configs – Final

### Router R1

```
R1# show run
```

```
Building configuration...
```

```
Current configuration : 2152 bytes
!
version 16.9
service timestamps debug datetime msec
service timestamps log datetime msec
platform qfp utilization monitor load 80
no platform punt-keepalive disable-kernel-core
!
hostname R1
!
boot-start-marker
boot-end-marker
!
no aaa new-model
!
no ip domain lookup
!
ip dhcp pool HOSTS
 network 192.168.1.0 255.255.255.0
 default-router 192.168.1.1
!
```



## Lab - Implement EIGRP for IPv4

---

```
login on-success log
!
subscriber templating
!
multilink bundle-name authenticated
!
!
key chain EIGRP-AUTHEN-KEY
  key 1
    key-string $3cre7!!
!
spanning-tree extend system-id
!
redundancy
  mode none
!
interface GigabitEthernet0/0/0
  ip address 10.0.12.1 255.255.255.0
  negotiation auto
!
interface GigabitEthernet0/0/1
  no ip address
  negotiation auto
!
interface GigabitEthernet0/0/1.1
  encapsulation dot1Q 1 native
  ip address 172.16.1.1 255.255.255.0
!
interface GigabitEthernet0/0/1.2
  encapsulation dot1Q 2
  ip address 192.168.1.1 255.255.255.0
!
interface Serial0/1/0
  no ip address
!
interface Serial0/1/1
  no ip address
!
router eigrp BASIC-EIGRP-LAB
!
  address-family ipv4 unicast autonomous-system 27
!
  af-interface GigabitEthernet0/0/1.2
    passive-interface
  exit-af-interface
!
  af-interface GigabitEthernet0/0/0
    authentication mode md5
    authentication key-chain EIGRP-AUTHEN-KEY
  exit-af-interface
```

## Lab - Implement EIGRP for IPv4

---

```
!  
af-interface GigabitEthernet0/0/1.1  
  authentication mode hmac-sha-256 $3cre7!!  
exit-af-interface  
!  
topology base  
exit-af-topology  
network 10.0.12.0 0.0.0.255  
network 172.16.1.0 0.0.0.255  
network 192.168.1.0  
eigrp router-id 1.1.1.1  
exit-address-family  
!  
ip forward-protocol nd  
no ip http server  
ip http secure-server  
!  
control-plane  
!  
banner motd ^C R1, Implement EIGRP for IPv4 ^C  
!  
line con 0  
  exec-timeout 0 0  
  logging synchronous  
  transport input none  
  stopbits 1  
line aux 0  
  stopbits 1  
line vty 0 4  
  exec-timeout 0 0  
  privilege level 15  
  password cisco123  
  login  
!  
end
```

### Router R2

```
R2# show run  
Building configuration...  
  
Current configuration : 1642 bytes  
!  
version 16.9  
service timestamps debug datetime msec  
service timestamps log datetime msec  
platform qfp utilization monitor load 80  
no platform punt-keepalive disable-kernel-core  
!
```

## Lab - Implement EIGRP for IPv4

---

```
hostname R2
!
boot-start-marker
boot-end-marker
!
no aaa new-model
!
no ip domain lookup
!
login on-success log
!
subscriber templating
!
multilink bundle-name authenticated
!
key chain EIGRP-AUTHEN-KEY
  key 1
    key-string $3cre7!!
!
spanning-tree extend system-id
!
redundancy
  mode none
!
interface GigabitEthernet0/0/0
  bandwidth 800000
  ip address 10.0.12.2 255.255.255.0
  ip authentication mode eigrp 27 md5
  ip authentication key-chain eigrp 27 EIGRP-AUTHEN-KEY
  negotiation auto
!
interface GigabitEthernet0/0/1
  bandwidth 800000
  ip address 10.0.23.2 255.255.255.0
  ip authentication mode eigrp 27 md5
  ip authentication key-chain eigrp 27 EIGRP-AUTHEN-KEY
  negotiation auto
!
router eigrp 27
  network 10.0.0.0 0.0.31.255
  passive-interface default
  no passive-interface GigabitEthernet0/0/0
  no passive-interface GigabitEthernet0/0/1
  eigrp router-id 2.2.2.2
!
ip forward-protocol nd
no ip http server
ip http secure-server
!
control-plane
```

## Lab - Implement EIGRP for IPv4

---

```
!  
banner motd ^C R2, Implement EIGRP for IPv4 ^C  
!  
line con 0  
  exec-timeout 0 0  
  logging synchronous  
  transport input none  
  stopbits 1  
line aux 0  
  stopbits 1  
line vty 0 4  
  exec-timeout 0 0  
  privilege level 15  
  password cisco123  
  login  
!  
end
```

### Router R3

```
R3# show run  
Building configuration...  
  
Current configuration : 1984 bytes  
!  
version 16.9  
service timestamps debug datetime msec  
service timestamps log datetime msec  
platform qfp utilization monitor load 80  
no platform punt-keepalive disable-kernel-core  
!  
hostname R3  
!  
boot-start-marker  
boot-end-marker  
!  
no aaa new-model  
!  
no ip domain lookup  
!  
login on-success log  
!  
subscriber templating  
!  
multilink bundle-name authenticated  
!  
key chain EIGRP-AUTHEN-KEY  
  key 1  
    key-string $3cre7!!
```

## Lab - Implement EIGRP for IPv4

---

```
!  
diagnostic bootup level minimal  
!  
spanning-tree extend system-id  
!  
redundancy  
mode none  
!  
interface Loopback0  
ip address 192.168.3.1 255.255.255.0  
!  
interface GigabitEthernet0/0/0  
ip address 10.0.23.3 255.255.255.0  
negotiation auto  
!  
interface GigabitEthernet0/0/1  
ip address 172.16.13.1 255.255.255.0  
negotiation auto  
!  
interface Serial0/1/0  
no ip address  
!  
interface Serial0/1/1  
no ip address  
!  
router eigrp BASIC-EIGRP-LAB  
!  
address-family ipv4 unicast autonomous-system 27  
!  
af-interface default  
passive-interface  
exit-af-interface  
!  
af-interface GigabitEthernet0/0/0  
authentication mode md5  
authentication key-chain EIGRP-AUTHEN-KEY  
no passive-interface  
exit-af-interface  
!  
af-interface GigabitEthernet0/0/1  
no passive-interface  
exit-af-interface  
!  
topology base  
variance 2  
exit-af-topology  
network 10.0.23.0 0.0.0.255  
network 172.16.13.0 0.0.0.255  
network 192.168.3.0  
eigrp router-id 3.3.3.3
```

## Lab - Implement EIGRP for IPv4

---

```
exit-address-family
!  
ip forward-protocol nd  
no ip http server  
ip http secure-server  
!  
control-plane  
!  
banner motd ^C R3, Implement EIGRP for IPv4 ^C  
!  
line con 0  
  exec-timeout 0 0  
  logging synchronous  
  transport input none  
stopbits 1  
line aux 0  
  stopbits 1  
line vty 0 4  
  exec-timeout 0 0  
  privilege level 15  
  password cisco123  
  login  
!  
end
```

### Switch D1

```
D1# show run  
Building configuration...  
  
Current configuration : 6666 bytes  
!  
version 16.9  
no service pad  
service timestamps debug datetime msec  
service timestamps log datetime msec  
! Call-home is enabled by Smart-Licensing.  
service call-home  
no platform punt-keepalive disable-kernel-core  
!  
hostname D1  
!  
vrf definition Mgmt-vrf  
!  
  address-family ipv4  
  exit-address-family  
!  
  address-family ipv6  
  exit-address-family  
!
```

## Lab - Implement EIGRP for IPv4

---

```
no aaa new-model
switch 1 provision ws-c3650-24td
!
no ip domain lookup
!
login on-success log
!
license boot level ipservicesk9
!
diagnostic bootup level minimal
!
spanning-tree mode rapid-pvst
spanning-tree extend system-id
!
redundancy
mode sso
!
transceiver type all
monitoring
!
class-map match-any system-cpp-police-topology-control
description Topology control
class-map match-any system-cpp-police-sw-forward
description Sw forwarding, L2 LVX data, LOGGING
class-map match-any system-cpp-default
description Inter FED, EWLC control, EWLC data
class-map match-any system-cpp-police-sys-data
description Learning cache ovfl, High Rate App, Exception, EGR Exception,
NFLSAMPLED DATA, RPF Failed
class-map match-any system-cpp-police-punt-webauth
description Punt Webauth
class-map match-any system-cpp-police-l2lvx-control
description L2 LVX control packets
class-map match-any system-cpp-police-forus
description Forus Address resolution and Forus traffic
class-map match-any system-cpp-police-multicast-end-station
description MCAST END STATION
class-map match-any system-cpp-police-multicast
description Transit Traffic and MCAST Data
class-map match-any system-cpp-police-l2-control
description L2 control
class-map match-any system-cpp-police-dot1x-auth
description DOT1X Auth
class-map match-any system-cpp-police-data
description ICMP redirect, ICMP_GEN and BROADCAST
class-map match-any system-cpp-police-stackwise-virt-control
description Stackwise Virtual
class-map match-any non-client-nrt-class
class-map match-any system-cpp-police-routing-control
description Routing control and Low Latency
```

## Lab - Implement EIGRP for IPv4

---

```
class-map match-any system-cpp-police-protocol-snooping
  description Protocol snooping
class-map match-any system-cpp-police-dhcp-snooping
  description DHCP snooping
class-map match-any system-cpp-police-system-critical
  description System Critical and Gold Pkt
!
policy-map system-cpp-policy
!
interface GigabitEthernet0/0
  vrf forwarding Mgmt-vrf
  no ip address
  negotiation auto
!
interface GigabitEthernet1/0/1
  switchport mode access
  spanning-tree portfast
!
interface GigabitEthernet1/0/2
  shutdown
!
interface GigabitEthernet1/0/3
  shutdown
!
interface GigabitEthernet1/0/4
  shutdown
!
interface GigabitEthernet1/0/5
  shutdown
!
interface GigabitEthernet1/0/6
  shutdown
!
interface GigabitEthernet1/0/7
  shutdown
!
interface GigabitEthernet1/0/8
  shutdown
!
interface GigabitEthernet1/0/9
  shutdown
!
interface GigabitEthernet1/0/10
  shutdown
!
interface GigabitEthernet1/0/11
  switchport mode trunk
!
interface GigabitEthernet1/0/12
  shutdown
```



## Lab - Implement EIGRP for IPv4

---

```
!  
interface GigabitEthernet1/0/13  
shutdown  
!  
interface GigabitEthernet1/0/14  
shutdown  
!  
interface GigabitEthernet1/0/15  
shutdown  
!  
interface GigabitEthernet1/0/16  
shutdown  
!  
interface GigabitEthernet1/0/17  
shutdown  
!  
interface GigabitEthernet1/0/18  
shutdown  
!  
interface GigabitEthernet1/0/19  
shutdown  
!  
interface GigabitEthernet1/0/20  
shutdown  
!  
interface GigabitEthernet1/0/21  
shutdown  
!  
interface GigabitEthernet1/0/22  
shutdown  
!  
interface GigabitEthernet1/0/23  
switchport access vlan 2  
switchport mode access  
spanning-tree portfast  
!  
interface GigabitEthernet1/0/24  
shutdown  
!  
interface GigabitEthernet1/1/1  
shutdown  
!  
interface GigabitEthernet1/1/2  
shutdown  
!  
interface TenGigabitEthernet1/1/3  
!  
interface TenGigabitEthernet1/1/4  
!  
interface Vlan1
```

## Lab - Implement EIGRP for IPv4

---

```
no ip address
!
ip forward-protocol nd
ip http server
ip http secure-server
!
control-plane
service-policy input system-cpp-policy
!
banner motd ^C D1, Implement EIGRP for IPv4 ^C
!
line con 0
exec-timeout 0 0
logging synchronous
stopbits 1
line aux 0
stopbits 1
line vty 0 4
exec-timeout 0 0
privilege level 15
password cisco123
login
line vty 5 15
login
!
end
```

### Switch D2

```
D2# show run
Building configuration...

Current configuration : 7011 bytes
!
version 16.9
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
! Call-home is enabled by Smart-Licensing.
service call-home
no platform punt-keepalive disable-kernel-core
!
hostname D2
!
vrf definition Mgmt-vrf
!
address-family ipv4
exit-address-family
!
address-family ipv6
```

## Lab - Implement EIGRP for IPv4

---

```
exit-address-family
!
no aaa new-model
switch 1 provision ws-c3650-24td
!
ip routing
!
no ip domain lookup
!
login on-success log
!
key chain EIGRP-AUTHEN-KEY
  key 1
    key-string $3cre7!!
!
license boot level ipservicesk9
!
diagnostic bootup level minimal
!
spanning-tree mode rapid-pvst
spanning-tree extend system-id
!
redundancy
  mode sso
!
transceiver type all
  monitoring
!
class-map match-any system-cpp-police-topology-control
  description Topology control
class-map match-any system-cpp-police-sw-forward
  description Sw forwarding, L2 LVX data, LOGGING
class-map match-any system-cpp-default
  description Inter FED, EWLC control, EWLC data
class-map match-any system-cpp-police-sys-data
  description Learning cache ovfl, High Rate App, Exception, EGR Exception,
NFLSAMPLED DATA, RPF Failed
class-map match-any system-cpp-police-punt-webauth
  description Punt Webauth
class-map match-any system-cpp-police-l2lvx-control
  description L2 LVX control packets
class-map match-any system-cpp-police-forus
  description Forus Address resolution and Forus traffic
class-map match-any system-cpp-police-multicast-end-station
  description MCAST END STATION
class-map match-any system-cpp-police-multicast
  description Transit Traffic and MCAST Data
class-map match-any system-cpp-police-l2-control
  description L2 control
class-map match-any system-cpp-police-dot1x-auth
```

## Lab - Implement EIGRP for IPv4

---

```
description DOT1X Auth
class-map match-any system-cpp-police-data
description ICMP redirect, ICMP_GEN and BROADCAST
class-map match-any system-cpp-police-stackwise-virt-control
description Stackwise Virtual
class-map match-any non-client-nrt-class
class-map match-any system-cpp-police-routing-control
description Routing control and Low Latency
class-map match-any system-cpp-police-protocol-snooping
description Protocol snooping
class-map match-any system-cpp-police-dhcp-snooping
description DHCP snooping
class-map match-any system-cpp-police-system-critical
description System Critical and Gold Pkt
!
policy-map system-cpp-policy
!
interface GigabitEthernet0/0
vrf forwarding Mgmt-vrf
no ip address
negotiation auto
!
interface GigabitEthernet1/0/1
no switchport
ip address 172.16.1.2 255.255.255.0
!
interface GigabitEthernet1/0/2
shutdown
!
interface GigabitEthernet1/0/3
shutdown
!
interface GigabitEthernet1/0/4
shutdown
!
interface GigabitEthernet1/0/5
shutdown
!
interface GigabitEthernet1/0/6
shutdown
!
interface GigabitEthernet1/0/7
shutdown
!
interface GigabitEthernet1/0/8
shutdown
!
interface GigabitEthernet1/0/9
shutdown
!
```

## Lab - Implement EIGRP for IPv4

---

```
interface GigabitEthernet1/0/10
shutdown
!
interface GigabitEthernet1/0/11
no switchport
ip address 172.16.13.2 255.255.255.0
!
interface GigabitEthernet1/0/12
shutdown
!
interface GigabitEthernet1/0/13
shutdown
!
interface GigabitEthernet1/0/14
shutdown
!
interface GigabitEthernet1/0/15
shutdown
!
interface GigabitEthernet1/0/16
shutdown
!
interface GigabitEthernet1/0/17
shutdown
!
interface GigabitEthernet1/0/18
shutdown
!
interface GigabitEthernet1/0/19
shutdown
!
interface GigabitEthernet1/0/20
shutdown
!
interface GigabitEthernet1/0/21
shutdown
!
interface GigabitEthernet1/0/22
shutdown
!
interface GigabitEthernet1/0/23
shutdown
!
interface GigabitEthernet1/0/24
shutdown
!
interface GigabitEthernet1/1/1
shutdown
!
interface GigabitEthernet1/1/2
```

## Lab - Implement EIGRP for IPv4

---

```
shutdown
!
interface TenGigabitEthernet1/1/3
!
interface TenGigabitEthernet1/1/4
!
interface Vlan1
no ip address
!
router eigrp BASIC-EIGRP-LAB
!
address-family ipv4 unicast autonomous-system 27
!
af-interface GigabitEthernet1/0/1
authentication mode hmac-sha-256 $3cre7!!
exit-af-interface
!
topology base
exit-af-topology
network 172.16.0.0
eigrp router-id 132.132.132.132
exit-address-family
!
ip forward-protocol nd
ip http server
ip http secure-server
!
control-plane
service-policy input system-cpp-policy
!
banner motd ^C D2, Implement EIGRP for IPv4 ^C
!
line con 0
exec-timeout 0 0
logging synchronous
stopbits 1
line aux 0
stopbits 1
line vty 0 4
exec-timeout 0 0
privilege level 15
password cisco123
login
line vty 5 15
login
!
end
```