Cisco Networking All-in-One

To create and configure a Cisco network, you need to know about routers and switches to develop and manage secure Cisco systems. Become acquainted with Cisco network devices and code listings; and find out how to manage static routing and view routing information.

OSI Model for Cisco Networking

While you may not use the OSI model every day, you should be familiar with it, specifically when working with Cisco switches and routers (which operate at Layer 2 and Layer 3, respectively). Here are some of the items that operate at each level of the OSI model:

Layer	Description	Examples
7. Application	Responsible for initiating or services the request.	SMTP, DNS, HTTP, and Telnet
6. Presentation	Formats the information so that it is understood by the receiving system.	Compression and encryption depending on the implementation
5. Session	Responsible for establishing, managing, and terminating the session.	NetBIOS
4. Transport	Breaks information into segments and is responsible for connection and connectionless communication.	TCP and UDP
3. Network	Responsible for logical addressing and routing	IP, ICMP, ARP, RIP, IGRP, and routers
2. Data Link	Responsible for physical addressing, error correction, and preparing the information for the media	MAC address, CSMA/CD, switches, and bridges
1. Physical	Deals with the electrical signal.	Cables, connectors, hubs, and repeaters

How to Configure a Cisco Network

Like all networks, a Cisco network needs to be properly configured. To do so, you need to know the configuration modes to use when configuring your network. You also should know how to configure an interface, configure a switch management interface, and configure an interface to use DHCP for your Cisco network.

Configuration modes for Cisco networking

When moving around in the Cisco IOS, you will see many prompts. These prompts change as you move from one configuration mode to another. Here is a summary of the major configuration modes:

- **User EXEC mode:** When you connect to a Cisco device the default configuration mode is user exec mode. With user exec mode you can view the settings on the device but not make any changes. You know you are in User EXEC mode because the IOS prompt displays a ">".
- **Privileged EXEC mode:** In order to make changes to the device you must navigate to Privileged EXEC mode where you may be required to input a password. Privileged EXEC mode displays with a "#" in the prompt.
- **Global Configuration mode:** Global Configuration mode is where you go to make global changes to the router such as the hostname. To navigate to Global Configuration mode from Privileged EXEC mode you type "configure terminal" or "conf t" where you will be placed at the "(config)#" prompt.
- **Sub Prompts:** There are a number of different sub prompts from Global Configuration mode you can navigate to such as the interface prompts to modify settings on a specific interface, or the line prompts to modify the different ports on the device.

Configure an interface for Cisco networking

When working with routers in particular, but also when dealing the management interface on switches, you will often need to configure network interfaces which will either match physical interface ports or virtual interfaces in the form of a virtual LAN (VLAN) interface (when dealing with switches).

For your router interfaces the following example will set speed, duplex and IP configuration information for the interface FastEthernet 0/0 (notice the interface reference as slot/port). In the case of the router, the interface is enabled using the no shutdown command in the final step; interfaces on switches are enabled by default.

```
Router1>enable
Router1#configure terminal
Router1(config)#interface FastEthernet0/0
Router1(config-if)#description Private LAN
Router1(config-if)#speed 100
Router1(config-if)#duplex full
Router1(config-if)#ip address 192.168.1.1 255.255.255.0
Router1(config-if)#no shutdown
```

Configure a switch management interface for Cisco networking

For your switches, to enable an IP address on your management interface, you will use something similar to this example. In this example, management is being performed over VLAN 1 - the default VLAN.

```
Switch1>enable
Switch1#configure terminal
Switch1#interface VLAN 1
Switch1(config-if)#ip address 192.168.1.241 255.255.255.0
```

Configure an interface to use DHCP for Cisco networking

If you want to configure either a router or switch to retrieve its IP configuration information from a network Dynamic Host Configuration Protocol (DHCP) server, then you can commands like the following example.

```
Router1>enable
Router1#configure terminal
Router1(config)#interface FastEthernet0/0
Router1(config-if)#ip dhcp
```

Creating a VLAN for Cisco Networking

When working with your Cisco network, you may want to separate users into different broadcast domains for security or traffic reduction. You can do this by implementing VLANs. The following example will create VLAN (VLAN2) and place the ports on a switch (from 1-12) into VLAN2.

```
Switch1>enable
Switch1#configure terminal
Switch1(config)#interface vlan 2
Switch1(config-if)#description Finance VLAN
Switch1(config-if)#exit
Switch1(config)#interface range FastEthernet 0/1 , FastEthernet 0/12
Switch1(config-if-range)#switchport mode access
Switch1(config-if-range)#switchport access vlan 2
```

If you are connecting two switches together, then you will want to allow all configured VLANs to pass between the two switches. This is accomplished by implementing a trunk port. To configure port 24 on your switch to be a trunk port, you will use the following code:

```
Switch1>enable
Switch1#configure terminal
Switch1(config)#interface FastEthernet 0/24
Switch1(config-if-range)#switchport mode trunk
```

Using EtherChannel for Cisco Networking

Don't be afraid to use EtherChannel on your Cisco network. EtherChannel allows you to take up to eight network ports on your switch and treat them as a single larger link. This can be used to connect servers with multiple network cards that are bonded (or teamed) to a switch, or to connect multiple switches together. There are two main negotiation protocols, Port Aggregation Protocol (PAgP) which is a proprietary Cisco protocol and Link Aggregation Control Protocol (LACP) which is an open standards protocol.

To set EtherChannel to use with of the protocols you will configure it to support one of the following modes.

- `auto`: Sets the interface to respond to PAgP negotiation packets, but the interface will start negotiations on its own.
- `desireable`: Sets the interface to actively attempt to negotiate a PAgP connection.
- `on`: Forces the connection to bring all links up without using a protocol to negotiate connections. This mode can only connect to another device that is also set to **on**. When using this mode, the switch does not negotiate the link using either PAgP or LACP.
- `active`: Sets the interface to actively attempt to negotiate connections with other LACP devices.
- `passive`: Sets the interface to respond to LACP data if it receives negotiation requests from other systems.

The following example will configure EtherChannel to use group ports 11 and 12 on the switch together using PAgP as the protocol. The same type of command would be used on the switch to which Switch1 is connected.

```
Switch1> enable
Switch1# configure terminal
Switch1(config)# interface range FastEthernet0/11 -12
Switch1(config-if-range)# switchport mode access
Switch1(config-if-range)# switchport access vlan 10
Switch1(config-if-range)# channel-group 5 mode desirable
```

Working with Spanning Tree Protocol for Cisco Networking

Spanning Tree Protocol (STP) enables you to create redundant loops on your Cisco network for fault tolerance, and prevents inadvertent loops that may be created on your network from bringing the network to its knees.

The following code will enable the Cisco proprietary Rapid Per VLAN Spanning Tree Protocol (PVST) over the open standard of Multiple Spanning Tree Protocol (MSTP). In addition to configuring STP on the switch, you will also configure port 2 on the switch for portfast, which allows the port to immediately transition to forwarding mode.

```
Switch1> enable
Switch1# configure terminal
Switch1(config)#spanning-tree mode rapid-pvst
Switch1(config)#interface FastEthernet 0/2
Switch1(config-if)#spanning-tree portfast
%Warning: portfast should only be enabled on ports connected to a single
 host. Connecting hubs, concentrators, switches, bridges, etc... to this
 interface when portfast is enabled, can cause temporary bridging loops.
 Use with CAUTION
%Portfast will be configured in 10 interfaces due to the range command
 but will only have effect when the interfaces are in a non-trunking mode.
```

Managing Static Routing for Cisco Networking

When working with your routers on your Cisco network, it's very likely that you'll want to have your routers route data. The first step in having your router pass data from one interface to another interface is to enable routing; just use these commands.

```
Router1>enable
Router1#configure terminal
Router1(config)#ip routing
```

Whether or not you choose to use a dynamic routing protocol, you may add static routes to your router. The following will add a static route to Router1 to send data to the 192.168.5.0/24 network using the router with the IP address of 192.168.3.2.

```
Router1>enable
Router1#configure terminal
Router1(config)#ip routing
Router1(config)#ip route 192.168.5.0 255.255.255.0 192.168.3.2
```

Managing routing information protocol for Cisco networking

Routing Information Protocol (RIP) is widely used, with version 2 allowing you to use Variable Length Subnet Masks (VLSM) across your network. The following code will enable routing, enable RIP, set RIP to version 2, disable route summarization, defines the distributed network from this router as 192.168.5.0/24, and rather than broadcasting routes, it will send RIP data directly to 192.168.1.1.

```
Router2>enable
Router2#configure terminal
Router2(config)#ip routing
Router2(config)#router rip
Router2(config-router)#version 2
Router2(config-router)#no auto-summary
Router1(config-router)#network 192.168.5.0
Router2(config-router)#neighbor 192.168.1.1
```

Managing enhanced interior gateway routing protocol for Cisco networking

Enhanced Interior Gateway Routing Protocol (EIGRP) is the updated version of IGRP. The following code will enable EIGRP using an autonomous-system (AS) number of 100, distribute two networks and disables auto summary.

```
Router2>enable
Router2#configure terminal
Router2(config)#ip routing
Router2(config)#router eigrp 100
Router2(config-router)#network 192.168.1.0
Router2(config-router)#network 192.168.5.0
Router2(config-router)#no auto-summary
```

Managing open shortest path first for Cisco networking

Open Shortest Path First (OSPF) is a link state protocol which is widely used. OSPF uses the address of the loopback interface as the OSPF identifier, so this example will set the address of the loopback interface, then enable OSPF with a process ID of 100, and distributing a network of 192.168.255.254 and a network of 192.168. 5.0/24

```
Router2>enable
Router2#configure terminal
```

```
Router2(config)#interface loopback 0
Router2(config-if)#ip address 192.168.255.254 255.255.255.0
Router2(config-if)#exit
Router2(config)#router ospf 100
Router2(config-router)#network 192.168.255.254 0.0.0.0 area 0
Router2(config-router)#network 192.168.5.0 0.0.0.255 area 0
```

Viewing Routing Information for Cisco Networking

After setting up any routing protocol that you want to implement - RIP, OSPF, or EIGRP - you can view all of your routing information through the `ip route`command. The following is an example of the output of this command. The output includes a legend showing the codes for each routing protocol, and the specific routes are identified by the source protocol.

```
Router2>enable
Password:
Router2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
D    192.168.10.0/24 [90/284160] via 192.168.1.1, 00:04:19, FastEthernet0/0
O    192.168.10.0/24 [110/11] via 192.168.1.1, 00:01:01, FastEthernet0/0
R    192.168.10.0/24 [120/1] via 192.168.1.1, 00:00:07, FastEthernet0/0
C    192.168.5.0/24 is directly connected, FastEthernet0/1
C    192.168.1.0/24 is directly connected, FastEthernet0/0
S    192.168.3.0/24 [1/0] via 192.168.1.1
```

Securing a Cisco Network

Security is always a concern, and your Cisco network needs to be properly secured. In the following sections, you see how to secure your Cisco network by configuring NAT, by configuring an ACL, and by applying that ACL.

Securing your Cisco network by configuring NAT

The following commands are used to configure NAT overload services on a router called Router1. In this example, a list of source address is created in access list #1, which is then used as the inside source list. The FastEthernet 0/0 port is the overloaded public address port that all inside addresses get translated to.

```
Router1>enable
Router1#configure terminal
Router1(config)#access-list 1 permit 10.0.0.0 0.255.255.255
Router1(config)#ip nat inside source list 1 interface FastEthernet 0/0 overload
Router1(config)#interface FastEthernet0/0
Router1(config-if)#ip nat outside
Router1(config-if)#interface FastEthernet0/1
Router1(config-if)#ip nat inside
```

Securing your Cisco network by configuring an access control list (ACL)

ACLs are used to control traffic flow. They can be used allow or deny the flow of traffic. The two main types of ACLs are:

- Standard ACLs, which have fewer **options** for classifying data and controlling traffic flow than Extended ACLs. They are only able to manage traffic based on the source IP address. These ACLs are numbered from 1–99 and from 1300–1999.
- Extended ACLs, which offer the ability to filter or control traffic based on a variety of criteria such as source or destination IP addresses, as well as protocol type such as, ICMP, TCP, UDP, or IP. These ACLs are numbered from 100–199 and from 2000–2699.

To create a standard ACL, you can use the following example which will create an ACL that allows traffic for the 192.168.8.0/24 network.

```
Switch1>enable
Switch1#configure terminal
Switch1(config)#access-list 50 permit 192.168.8.0 0.0.0.255
```

To create an extended ACL you can use the following example which will create an ACL that allows traffic with addresses in the 192.168.8.0/24 network and tcp ports of either 80 (http) or 443 (https):

```
Router1>enable
Router1#configure terminal
Router1(config)#access-list 101 remark This ACL is to control the outbound router traffic.
Router1(config)#access-list 101 permit tcp 192.168.8.0 0.0.0.255 any eq 80
Router1(config)#access-list 101 permit tcp 192.168.8.0 0.0.0.255 any eq 443
```

Securing your Cisco network by applying an access control list

After you have created an Access Control List (ACL), such as ACL 101 created above, you can apply that ACL to an interface. In the following example, this ACL is placed to restrict outbound traffic on FastEthernet0/1.

```
Router1>enable
Router1#configure terminal
Router1(config)#interface FastEthernet0/1
Router1(config-if)#ip access-group 101 out
```

PORT SECURITY

```
Switch>enable
Password: cisco
Switch#show running-config
Switch#configure terminal
Switch(config)#interface fa0/12
Switch(config-if)#switchport mode access
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security maximum 2
Switch(config-if)#switchport port-security violation shutdown
Switch(config-if)#no shutdown
Switch(config-if)#end
Switch#show port-security interface fa0/12
Switch#copy running-config startup-config
```

Cisco Access Control Lists:

Standard ACL: 1 – 99 and 1300 – 1999

- Use a remark to describe the ACL (Optional):

```
1  R1(config)# access-list 1 remark ACL TO DENY
```

ACCESS FROM SALES VLAN

- Create the ACL, keeping the following in mind:
  - ACL uses first-match logic.
  - There is an implicit deny anyat the end of the ACL.

```
R1(config)# access-list 2 deny 192.168.1.77
R1(config)# access-list 2 deny 192.168.1.64
0.0.0.31
R1(config)# access-list 2 permit 10.1.0.0
0.0.255.255
R1(config)# access-list 2 deny 10.0.0.0
0.255.255.255
R1(config)# access-list 2 permit any
```

- Enable the ACL on the chosen router interface in the correct direction (in or out):

```
R1(config-if)# ip access-group 2 out
```

- Using standard ACL to limit telnet and SSH access to a router:

Create the ACL that defines the permitted telnet clients:

```
R1(config)# access-list 99 remark ALLOWED TELNET
CLIENTS
R1(config)# access-list 99 permit 192.168.1.128
0.0.0.15
```

Apply the ACL inbound the vty lines

```
R1(config)# line vty 0 4
R1(config-line)# access-class 99 in
```

Extended ACL: 100 – 199 and 2000 – 2699

- Extended ACL should be placed as close as possible to the source of the packet.
- Extended ACL matches packets based on source & des.IP addresses, protocol, source & des. Port numbers andother criteria as well

```
R1(config)# access-list 101 remark MY_ACCESS_LIST
R1(config)# access-list 101 deny iphost 10.1.1.1
host 10.2.2.2
R1(config)# access-list 101 deny tcp 10.1.1.0
0.0.0.255 any eq 23
R1(config)# access-list 101 deny icmp 10.1.1.1
0.0.0.0 any
R1(config)# access-list 101 deny tcphost 10.1.1.0
host 10.0.0.1 eq 80
R1(config)# access-list 101 deny udphost 10.1.1.7
eq 53 any
R1(config)# access-list 101 permit ip any any
R1(config)# interface fastEthernet 0/0
R1(config-if)# ip access-group 101 in
```

Named ACL:

- Named ACLs use names to identify ACLs rather than numbers, and commands that permit or deny traffic are written in a sub mode called named ACL mode (nacl).
- Named ACL enables the editing of the ACL (deleting or inserting statements) by sequencing statements of the ACL.
- Named standard ACL:

```
R1(config)# ip access-list standard
MY_STANDARD_ACL
R1(config-std-nacl)# permit 10.1.1.0 0.0.0.255
R1(config-std-nacl)# deny 10.2.2.2
R1(config-std-nacl)# permit any
R1(config)# interface fastEthernet 0/1
R1(config-if)# ip access-group MY STANDARD ACL out
```

- Named extended ACL:

```
R1(config)# ip access-list extended
MY_EXTENDED_ACL
R1(config-ext-nacl)# deny icmp 10.1.1.1 0.0.0.0
any
R1(config-ext-nacl)# deny tcphost 10.1.1.0 host
10.0.0.1 eq 80
R1(config-ext-nacl)# permit ip any any
R1(config)# interface fastEthernet 0/1
R1(config-if)# ip access-group MY EXTENDED ACL in
```

- Editing ACL using sequence numbers:

```
R1(config)# ip access-list extended
MY_EXTENDED_ACL
R1(config-ext-nacl)# no 20    ! Deletes the
statement of sequence number 20
R1(config)# ip access-list standard 99
R1(config-std-nacl)# 5 deny 1.1.1.1    ! inserts a
statement with sequence 5
```