CISCO IOS Command Reference for CCNA Exploration Semester 2

Basic configuration commands for router and switch

Router/Switch Modes:

- **Router>**: User mode = Limited to basic monitoring commands
- **Router#**: Privileged mode (exec-level mode) = Provides access to all other router commands
- **Router(config)#**: global configuration mode = Commands that affect the entire system
- **Router(config-if)#**: interface mode = Commands that affect interfaces
- **Router(config-subif)#**: subinterface mode = Commands that affect subinterfaces
- **Router(config-line)#**: line mode = Commands that affect in lines modes (console, vty, aux…)
- **Router(config-router)#**: router configuration mode

Changing switch hostname:

```
Switch(config)# hostname SW1
```

Configuring passwords:

```
SW1(config)# enable secret cisco      ! MD5 hash
SW1(config)# enable password notcisco     ! Clear text
```

Securing console port:

```
SW1(config)# line con 0
SW1(config-line)# password cisco
SW1(config-line)# login
```

Securing terminal lines:

```
SW1(config)# line vty 0 4
SW1(config-line)# password cisco
SW1(config-line)# login
```

Encrypting passwords:

```
SW1(config)# service password-encryption
```

Configuring banners:

```
SW1(config)# banner motd $ UNAUTHORIZED ACCESS ISPROHIBITED$
```

CISCO IOS Command Reference for CCNA Exploration Semester 2

Giving the switch an IP address:

```
SW1(config)# interface vlan 1
SW1(config-if)# ip address 172.16.1.11 255.255.255.0    ! or DHCP
SW1(config-if)# no shutdown
```

Setting the default gateway:

```
SW1(config)# ip default-gateway 172.16.1.1
```

Saving configuration:

```
SW1# copy running-config startup-config
```

Configuring switch to use SSH:

- Configure DNS domain name:

```
SW1(config)# ip domain-name example.com
```

- Configure a username and password:

```
SW1(config)# username admin password cisco
```

- Generate encryption keys:

The size of the key modulus in the range of 360 to 2048

```
SW1(config)# crypto key generate rsa
How many bits in the modulus [512]: 1024
```

- Define SSH version to use:

```
SW1(config)# ip ssh version 2
```

- Enable vty lines to use SSH:

```
SW1(config)# line vty 0 4
SW1(config-line)# login local
! You can set vty lines to use only telnet or only ssh or both as in the example.
SW1(config-line)# transport input telnet ssh
```

CISCO IOS Command Reference for CCNA Exploration Semester 2

Show and Verify Basic Configuration:

- Shows information about the switch and its interfaces, RAM, NVRAM, flash, IOS, etc.

```
SW1# show version
```

- Shows the current configuration file stored in DRAM.

```
SW1# show running-config
```

- Shows the configuration file stored in NVRAM which is used at first boot process.

```
SW1# show startup-config
```

- Lists the commands currently held in the history buffer.

```
SW1# show history
```

- Shows an overview of all interfaces, their physical status, protocol status and ip address if assigned.

```
SW1# show ip interface brief
```

- Shows detailed information about the specified interface, its status, protocol, duplex, speed, encapsulation, last 5 min traffic.

```
SW1# show interface vlan 1
```

- Shows the description of all interfaces

CISCO IOS Command Reference for CCNA Exploration Semester 2

Port Security and VLAN Commands

<mark>Configuring port security:</mark>

- Make the switch interface as access port:

```
SW1(config-if)# switchport mode access
```

- Enable port security on the interface:

```
SW1(config-if)# switchport port-security
```

- Specify the maximum number of allowed MAC addresses:

```
SW1(config-if)# switchport port-security maximum 1
```

- Define the action to take when violation occurs:

```
SW1(config-if)# switchport port-security violation shutdown    ! options:
shutdown, protect, restrict
```

- Specify the allowed MAC addresses:

The sticky keyword is used to let the interface dynamically learns and configures the MAC addresses of the currently connected hosts.

```
SW1(config-if)# switchport port-security mac-address 68b5.9965.1195    !
options: H.H.H, sticky
```

Verify and troubleshoot port security:

- Shows the entries of the mac address table:

```
SW1# show mac-address-table
```

- Overview of port security of all interfaces:

```
SW1# show port-security
```

- Shows detailed information about port security on the specified interface:

```
SW1# show port-security interface fa0/5
```

CISCO IOS Command Reference for CCNA Exploration Semester 2

Configuring VLANs:

- Create a new VLAN and give it a name:

```
SW1(config)# vlan 10
SW1(config-vlan)# name SALES
```

- Assign an access interface to access a specific VLAN:

```
SW1(config)# interface fastEthernet 0/5
SW1(config-if)# switchport mode access
SW1(config-if)# switchport access vlan 10
```

Configuring Trunks:

```
SW1(config)# interface fastEthernet 0/1
SW1(config-if)# switchport mode trunk    ! options: access, trunk, dynamic
auto, dynamic desirable
```

- Configure allowed VLANs for the trunk:

```
SW1(config-if)# switchport trunk allowed vlan 10,20,30
```

- Configure native VLAN for the trunk:

```
SW1(config-if)# switchport trunk native vlan 10
```

Securing VLANs and Trunking:

- Administratively disable unused interfaces:

```
SW1(config-if)# shutdown
```

- Prevent trunking by disabling auto negotiation on the interface:

```
SW1(config-if)# nonegotiate    ! or hardcode the port asan access port
SW1(config-if)# switchport mode access
```

Configuring router interfaces:

Clock rate is set only on the DCE side, typically the ISP side. On your router which is DTE you don't need to set clocking.

```
R1(config)# interface fastEthernet 0/0
R1(config-if)# description LINK TO LOCAL LAN THROUGH SW1
R1(config-if)# ip address 172.16.1.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
```

```
R1(config)# interface serial 0/1/0
R1(config-if)# description WAN CONNECTION TO R2
R1(config-if)# ip address 10.1.1.1 255.255.255.252
R1(config-if)# clock rate 128000
R1(config-if)# no shutdown
```

Configuring Router-On-Stick for vlan routing:

```
R1(config)# interface fastEthernet 0/0
R1(config-if)# no shutdown
R1(config)# interface fastEthernet 0/0.10
R1(config-subif)# encapsulation dot1q 10
R1(config-subif)# ip address 192.168.10.1 255.255.255.0
R1(config-subif)# interface fastEthernet 0/0.20
R1(config-subif)# encapsulation dot1q 20
R1(config-subif)# ip address 192.168.20.1 255.255.255.0
```

Verify and troubleshoot VLANs and VTP:

- Lists information about administrative setting and operation status of interface:

```
SW1# show interfaces if switchport
```

- Lists all the trunk ports on a switch including the trunk allowed VLANs:

```
SW1# show interfaces trunk
```

- Lists information about the VLANs:

```
SW1# show vlan {brief | id | name | summary}
```

CISCO IOS Command Reference for CCNA Exploration Semester 2

Routing Commands: Static, RIP, and OSPF

Static route:

- Using next hop:

```
R1(config)# ip route 10.1.2.0 255.255.255.0 10.1.128.1
```

- Using exit interface:

```
R1(config)# ip route 10.1.2.0 255.255.255.0 Serial 0/0
*Note: Exit interface can be used in point-to-point serial links.
```

Default Route using next hop:

```
R1(config)# ip route 0.0.0.0 0.0.0.0 199.1.1.1
```

Default Route using exit interface:

```
R1(config)# ip route 0.0.0.0 0.0.0.0 Serial 0/0
```

RIPv2 Configuration:

```
R1(config)# router rip
R1(config-router)# version 2
R1(config-router)# network 10.0.0.0    ! written as an original class A
R1(config-router)# no auto-summary
R1(config-router)# passive-interface serial 0/0
R1(config-router)# default-information originate
```

RIPv2 Verification:

- Shows information about the running routing protocol process:

```
R1# show ip protocols
```

- Shows the entire routing table:

```
R1# show ip route
```

- Shows routes learned via RIP only:

```
R1# show ip route rip
```

- Shows detailed information about the route to the specified destination network:

```
R1# show ip route 10.1.1.1
```

OSPF Configuration:

- Enter OSPF router configuration mode:

```
R1(config)# router ospf 10    ! 10 = process ID
```

- Configure one or more network commands to identify which interfaces will run OSPF:

```
R1(config-router)# network 10.0.0.0 0.255.255.255 area 0
R1(config-router)# network 172.16.8.0 0.0.7.255 area 0
R1(config-router)# network 192.168.1.254 0.0.0.0 area 0
```

- Distribute default route (Optional):

```
R1(config)# router ospf 10
R1(config-router)# default-information originate
```

- Disabling OSPF on a certain interface (Optional):

```
R1(config-router)# passive-interface serial 0/0
```

Configure router ID either for OSPF (Optional)

- Using router-id ospf subcommand:

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

- Configuring an IP address on a loopback interface:

```
R1(config)# interface loopback 0
R1(config-if)# ip address 1.1.1.1 255.255.255.255
```

Change Hello and Dead intervals per interface (Optional):

```
R1(config-if)# ip ospf hello-interval 2
R1(config-if)# ip ospf dead-interval 6
```

Impact routing choices by tuning interface cost using one of the following ways (Optional):

- Changing interface cost:

```
R1(config-if)# ip ospf cost 55
```

- Changing interface bandwidth:

```
R1(config-if)# bandwidth 128    ! in Kbps
```

- Changing the reference bandwidth that used by OSPF to calculate the cost:

```
R1(config-router)# auto-cost reference-bandwidth 1000    ! in Mbps
```

OSPF verification:

- Shows information about the running routing protocol process:

```
R1# show ip protocols
```

- Shows the entire routing table:

```
R1# show ip route
```

- Shows routes learned via OSPF only:

```
R1# show ip route ospf
```

- Shows all neighboring routers along with their respective adjacency state:

```
R1# show ip ospf neighbors
```

- Shows all the information contained in the LSDB:

```
R1# show ip ospf database
```

- Shows detailed information about OSPF running on a specific interface:

```
R1# show ip ospf interfaces serial 0/0
```

CISCO IOS Command Reference for CCNA Exploration Semester 2

ACL Commands

Access Control Lists:

Numbered Standard ACL: 1 – 99 and 1300 – 1999

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

```
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 any at the end of the ACL.

```
R1(config)# access-list 2 deny host 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
```

Numbered 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 & destination IP addresses, protocol, source & des. Port numbers and other criteria as well
- Configure ACL

```
R1(config)# access-list 101 remark MY_ACCESS_LIST
R1(config)# access-list 101 deny ip host 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 tcp host 10.1.1.0 host 10.0.0.1 eq 80
R1(config)# access-list 101 deny udp host 10.1.1.7 any eq 53
R1(config)# access-list 101 permit ip any any
```

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

```
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).

CISCO IOS Command Reference for CCNA Exploration Semester 2

- 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 host 10.2.2.2
R1(config-std-nacl)# permit any
```

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

```
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 tcp host 10.1.1.0 host 10.0.0.1 eq 80
R1(config-ext-nacl)# permit ip any any
```

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

```
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
```

Verifying ACLs:

- Shows all ACLs configured on a router with counters at the end of each statement:

```
R1# show access-lists

! OR

R1# show ip access-list
```

- Shows only the specified ACL:

```
R1# show ip access-list 101
```

CISCO IOS Command Reference for CCNA Exploration Semester 2

DHCP Commands

DHCP Server

- Enable DHCP, which is enabled by default.

```
R1(config)# service dhcp
```

- Define a DHCP pool and give it a name:

```
R1(config)# ip dhcp pool MY_POOL
```

- Define network and mask to use in this pool, the default gateway, and DNS servers:

```
R1(dhcp-config)# network 192.168.1.0 255.255.255.0
R1(dhcp-config)# default-router 192.168.1.1
R1(dhcp-config)# dns-server 213.131.65.20 8.8.8.8
```

- Confine the lease time (OPTIONAL):

```
R1(dhcp-config)#lease 2     ! Days
```

- Define one or more scopes of excluded (reserved) addresses (OPTIONAL):

```
R1(config)# ip dhcp excluded-address 192.168.1.1 192.168.1.100
R1(config)# ip dhcp excluded-address 192.168.1.200 192.168.1.254
```

- Configure router as a DHCP relay agent to forward DHCP packets to a DHCP server:

```
Router(config)#interface fastethernet 0/0
Router(config-if)#ip helper-address 172.16.20.2
```

DHCP Verification and Troubleshooting:

- Shows the status of the specified pool and the leased addresses from that pool:

```
R1# show ip dhcp pool MY_POOL
```

- Shows all the leased ip addresses from all configured DHCP pools:

```
R1# show ip dhcp binding
```

CISCO IOS Command Reference for CCNA Exploration Semester 2

NAT Commands

Network Address Translation (NAT):

Static NAT:

- Configure static NAT statement:

```
R1(config)# ip nat inside source static 192.168.1.10 200.1.1.1
```

- Define the outside and inside interfaces:

```
R1(config)# interface serial 0/0
R1(config-if)# ip nat outside
R1(config)# interface FastEthernet 1/1
R1(config-if)# ip nat inside
```

Dynamic NAT:

- Create an ACL that determines the IP addresses thatare allowed to be translated:

```
R1(config)# access-list 3 permit 192.168.1.0 0.0.0.255
```

- Create a pool of public IP addresses:

```
R1(config)# ip nat pool PUB 200.1.1.1 200.1.1.6 netmask 255.255.255.248
```

- Configure NAT statement:

```
R1(config)# ip nat inside source list 3 pool PUB
```

- Configure NAT with overload

```
R1(config)# ip nat inside source list 3 pool PUB overload
```

- Configure NAT with overload using interface

```
R1(config)# ip nat inside source list 3 interface s0/0/0 overload
```

- Define the outside and inside interfaces:

```
R1(config)# interface serial 0/0
R1(config-if)# ip nat outside
R1(config)# interface FastEthernet 1/1
R1(config-if)# ip nat inside
```

CISCO IOS Command Reference for CCNA Exploration Semester 2

NAT verification and troubleshoot:

- Useful in viewing the configuration of NAT pool and the inside and outside interfaces:

```
R1# show running-config
```

- Displays access lists, including the one used for NAT:

```
R1# show access-lists
```

- Shows counters for packets and NAT table entries, as well as basic configuration information:

```
R1# show ip nat stasitics
```

- Displays the NAT table:

```
R1# show ip nat translations
```

- Clears all the dynamic entries in the NAT table:

```
R1# clear ip nat translations *
```