

CCNA Lab Workbook

CCNA Routing and Switching



Mohamed Ouamer

COPYRIGHT INFORMATION

This publication, **CCNA LAB WORKBOOK SAMPLE LABS**, was developed by Mohamed Ouamer. All rights reserved. No part of this publication may be reproduced or distributed in any form or by any means without the prior written permission of Mohamed Ouamer.

Cisco®, Cisco Systems®, CCDA®, CCNA®, CCDP®, CCNP®, CCIE®, CCSI®, the Cisco Systems logo and the CCIE logo are trademarks or registered trademarks of Cisco Systems, Inc. in the United States and certain other countries. All other trademarks are trademarks of their respective owners.

If you have questions or concerns about our Copyright information, please email us at contact@networkexperttraining.com.

DISCLAIMER

The following publication, **CCNA LAB WORKBOOK SAMPLE LABS**, is developed to assist candidates in the preparation for Cisco Systems' CCNA exam.

While every effort has been made to ensure that all material is as complete and accurate as possible, the enclosed material is presented on an "as is" basis. Neither the authors nor networkexperttraining.com assume any liability or responsibility to any person or entity with respect to loss or damages incurred from the information contained in this workbook.

If you have questions or concerns about our disclaimer, please email us at contact@networkexperttraining.com.

TABLE OF CONTENT:

LAB 1: CDP.....	4
LAB 2: STATIC IP ROUTING	14
LAB 3: PORT SECURITY	28

II. Initial Configuration:

R1:

```
hostname R1
!
!
!
interface Loopback0
 ip address 1.1.1.1 255.255.255.0
!
interface FastEthernet0/0
 ip address 192.168.1.1 255.255.255.0
 no shutdown
 end
!
line con 0
 exec-timeout 0 0
 logging synchronous
 privilege level 15
!
line vty 0 4
 password cisco
 login
!
end
```

III. Tasks:

1. Set the hostname of the switch to "Rck9_SW1"
2. Set the enable password to swcisco
3. Set the console password to cnslcisco
4. Create the following vlans:

VLAN ID	VLAN NAME
10	VLAN_A
20	VLAN_B
30	VLAN_C

5. Add the port F0/1 to the vlan VLAN_A.
6. Add the port F0/2 to the vlan VLAN_B.
7. Add the port F0/3 to the vlan VLAN_C.

8. Display the MAC addresses learned by the switch dynamically.
9. Set the IP address of the interface vlan10 to 192.168.1.2/24.
10. Configure SW1 to route all IP traffic to R1.
11. Set the speed of the interface F0/1 to 10Mbps.
12. Set the speed of the interface F0/2 to 100Mbps, and the duplex mode to full.
13. Configure the CAM table according to the following table:

MAC Address	VLAN	Interface	Type
CCF4.1256.2559	10	FastEthernet0/24	static
CC0D.EF01.2345	10	FastEthernet0/23	static
CC94.1256.2559	10	FastEthernet0/22	static
CC84.1256.2559	10	FastEthernet0/21	static
CC74.1256.2559	20	FastEthernet0/20	static
CC64.1256.2559	20	FastEthernet0/20	static
CC54.1256.2559	20	FastEthernet0/20	static
CC44.1256.2559	20	FastEthernet0/20	static
CC34.1256.2559	30	FastEthernet0/19	static
CC24.1256.2559	30	FastEthernet0/18	static
CC14.1256.2559	30	FastEthernet0/17	static
CC04.1256.2559	30	FastEthernet0/18	static

14. Set the description of the interfaces as follows:

Interface	Description
F0/1	Connected To PC 1
F0/2	Connected To R1
F0/3	Connected To PC 2

IV. Solutions:

Task 1:

SW1:

```
Switch>enable
Switch#configure terminal
Switch(config)#hostname Rck9_SW1
Rck9_SW1(config)#end
Rck9_SW1#
```

In SW1's global configuration mode, type **hostname Rck9_SW1** to change the hostname of the switch. The default hostname of Cisco switches is "Switch".

The hostname command is used to assign a locally significant name to the switch. This command has the same effect on any Cisco router. Additionally, the hostname attribute is used during the PPP authentication process.

Task 2:

To accomplish this task, we have two choices:

1. Using the enable password command:

This statement configures the privileged level password. The password will be stored in clear text format as shown in the output of show running-config command below. You can encrypt the password using the service password-encryption command. But,

SW1:

```
Rck9_SW1#configure terminal
Rck9_SW1(config)#enable password swcisco
Rck9_SW1(config)#end
Rck9_SW1#
```

To verify your configuration, display the running-config file to see that the password has been accepted. Then log out from the console, and log in again.

Verification:

```
Rck9_SW1#show running-config
Building configuration...
```

```
Current configuration : 951 bytes
!
!
```

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
!
hostname Rck9_SW1
!
no logging console
!
enable password swcisco
```

```
Rck9_SW1#disable // Exit from the global configuration mode to the user mode
Rck9_SW1>enable // Enter the privileged mode
Password: // type the enable password
Rck9_SW1#
```

2. Using the enable secret command:

This statement assigns the privileged level password. The password will be stored in encrypted text format using the MD5 algorithm as shown in the output of show running-config command below.

SW1:

```
Rck9_SW1#configure terminal
Rck9_SW1(config)#enable secret swcisco
Rck9_SW1(config)#end
Rck9_SW1#
```

To verify your configuration, display the running-config file to see that the password has been accepted. Then log out from the console, and log in again.

Verification:

```
Current configuration : 1000 bytes
!
!
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
!
hostname Rck9_SW1
```

```

!
no logging console
!
enable secret 5 $1$TMLI$GivwhpLcLD1RgCwCB2DkZ0
enable password swcisco

```

```

Rck9_SW1#disable // Exit from the global configuration mode to the user mode
Rck9_SW1>enable // Enter the privileged mode
Password: // Type the enable password
Rck9_SW1#

```

If these two command are both configured, the enable secret command takes precedence over the enable password. It is recommend using the second solution because it is more secure than the first one. I will show you later how to decrypt the password encrypted using the service password-encryption command.

Task 3:

To configure the console password, follow these steps:

Step 1: In the global configuration mode enter the line configuration mode by typing the **line con 0** command.

Step 2: configure the password using the statement **password cnslcisco**

Step 3: use the login command to authenticate users. If the login command is not applied to the console line, the switch will not prompt you to enter the console password. So you can access the switch without being authenticated.

SW1:

```

Rck9_SW1#configure terminal
Rck9_SW1(config)#line con 0
Rck9_SW1(config-line)#password cnslcisco
Rck9_SW1(config-line)#login
Rck9_SW1(config-line)#^Z
Rck9_SW1#

```

Verification:

Now, let's log out from the switch's console, and then log in again.

```

Rck9_SW1#disable // Exit from the privileged EXEC mode to the user mode

```

```

Rck9_SW1>exit // Exit from the EXEC user mode

```

Rck9_SW1 con0 is now available

Press RETURN to get started.

// Press RETURN and type the console password

User Access Verification

Password:

Rck9_SW1>enable **// Enter the privileged mode**

Password: **// Type the enable password**

Rck9_SW1#

Task 4:

To create a vlan you have either to use the vlan database mode or the vlan configuration mode.

1. Using the vlan database mode:

```
Rck9_SW1>enable
Rck9_SW1#Vlan database
Rck9_SW1(vlan)#vlan 10 name VLAN_A
Rck9_SW1(vlan)#vlan 20 name VLAN_B
Rck9_SW1(vlan)#vlan 30 name VLAN_C
Rck9_SW1(vlan)#exit
APPLY completed.
Exiting....
Rck9_SW1#
```

Verification:

SW1:

Rck9_SW1#show vlan

VLAN	Name	Status	Ports
1	default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24

Gig1/1, Gig1/2

```

10 VLAN_A          active
20 VLAN_B          active
30 VLAN_C          active

```

<omitted output>

By default all the switch ports belong to vlan 1. Additionally you cannot change the name of vlan 1.

2. Using the vlan configuration mode:

SW1:

```

Rck9_SW1#configure terminal
Rck9_SW1(config)#vlan 10
Rck9_SW1(config-vlan)#name VLAN_A
Rck9_SW1(config-vlan)#vlan 20
Rck9_SW1(config-vlan)#name VLAN_B
Rck9_SW1(config-vlan)#vlan 30
Rck9_SW1(config-vlan)#name VLAN_C
Rck9_SW1(config-vlan)#end
Rck9_SW1#

```

Verification:

SW1:

Rck9_SW1#show vlan

VLAN	Name	Status	Ports
1	default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gig1/1, Gig1/2

```

10 VLAN_A          active
20 VLAN_B          active
30 VLAN_C          active

```

<omitted output>

Task 5:

To assign a switch port to a vlan, you have to use the **switchport access vlan** command at the port configuration mode.

SW1:

```
Rck9_SW1#configure terminal
Rck9_SW1(config)#interface fastethernet0/1
Rck9_SW1(config-if)#switchport access vlan 10
Rck9_SW1(config-if)#^Z
Rck9_SW1#
```

Verification:

SW1:

```
Rck9_SW1#show interfaces fastEthernet 0/1 switchport
Name: Fa0/1
Switchport: Enabled
Administrative Mode: dynamic auto
Operational Mode: static access
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
Negotiation of Trunking: On
Access Mode VLAN: 10 (VLAN_A)
Trunking Native Mode VLAN: 1 (default)
<omitted output>
```

The administrative mode parameter shows the default mode of the port. In our case, it is set to dynamic auto, which means that it will not initiate trunk link negotiation. Instead, if the mode of the neighbor interface is set to desirable or trunk, the port becomes trunk link the DTP negotiation. If not the port will behave as an access port.

The operational mode parameter shows you the actual running mode, which is access mode. The access mode vlan parameter shows you the VLAN to which the port will belong if the operation mode is set to static access. In our case this is VLAN 10.

```
Rck9_SW1#show vlan brief
```

VLAN Name	Status	Ports
1 default	active	Fa0/2, Fa0/3, Fa0/4, Fa0/5 Fa0/6, Fa0/7, Fa0/8, Fa0/9 Fa0/10, Fa0/11, Fa0/12, Fa0/13 Fa0/14, Fa0/15, Fa0/16, Fa0/17 Fa0/18, Fa0/19, Fa0/20, Fa0/21 Fa0/22, Fa0/23, Fa0/24, Gig1/1 Gig1/2
10 VLAN_A	active	Fa0/1
20 VLAN_B	active	
30 VLAN_C	active	

<omitted output>

As shown in the output of the show vlan brief command, the port F0/1 is now associated with VLAN 10. The show vlan command displays the ports associated with each vlan. Trunk ports are not associated with any vlan. To display the vlans allowed on a specific interface use the show interfaces trunk command.

Task 6:

SW1:

```
Rck9_SW1#configure terminal
Rck9_SW1(config)#interface fastethernet0/2
Rck9_SW1(config-if)#switchport access vlan 20
Rck9_SW1(config-if)#^Z
Rck9_SW1#
```

Verification:

SW1:

```
Rck9_SW1#show vlan brief
```

VLAN Name	Status	Ports
1 default	active	Fa0/3, Fa0/4, Fa0/5, Fa0/6 Fa0/7, Fa0/8, Fa0/9, Fa0/10 Fa0/11, Fa0/12, Fa0/13, Fa0/14 Fa0/15, Fa0/16, Fa0/17, Fa0/18

```

Fa0/19, Fa0/20, Fa0/21, Fa0/22
Fa0/23, Fa0/24, Gig1/1, Gig1/2
10 VLAN_A          active Fa0/1
20 VLAN_B          active Fa0/2
30 VLAN_C          active
<omitted output>

```

Task 7:**SW1:**

```

Rck9_SW1#configure terminal
Rck9_SW1(config)#interface fastethernet0/3
Rck9_SW1(config-if)#switchport access vlan 30
Rck9_SW1(config-if)#^Z
Rck9_SW1#

```

Verification:**SW1:**

```
Rck9_SW1#show vlan brief
```

VLAN Name	Status	Ports
1 default	active	Fa0/4, Fa0/5, Fa0/6, Fa0/7 Fa0/8, Fa0/9, Fa0/10, Fa0/11 Fa0/12, Fa0/13, Fa0/14, Fa0/15 Fa0/16, Fa0/17, Fa0/18, Fa0/19 Fa0/20, Fa0/21, Fa0/22, Fa0/23 Fa0/24, Gig1/1, Gig1/2
10 VLAN_A	active	Fa0/1
20 VLAN_B	active	Fa0/2
30 VLAN_C	active	Fa0/3

```
<omitted output>
```

Task 8:

To display all the MAC addresses learned by the switch, use the **show mac address-table** command. Using this command with the dynamic keyword will display only MAC addresses learned dynamically.

SW1:

```
Rck9_SW1#show mac address-table dynamic
Mac Address Table
```

```
-----
Vlan  Mac Address      Type    Ports
----  -
1     0050.56c0.0001    DYNAMIC Fa0/20
1     0200.4c4f.4f50    DYNAMIC Fa0/10
10    000f.23d5.5220    DYNAMIC Fa0/1
```

The CAM table has displays four information about each MAC address, which are the vlan ID, the value of MAC address, the type (informs you how the MAC address was learned), and the port over which the MAC address was learned.

Task 9:**SW1:**

```
Rck9_SW1#configure terminal
Rck9_SW1(config)#interface vlan 10
Rck9_SW1(config-if)#ip address 192.168.1.2 255.255.255.0
Rck9_SW1(config-if)#end
Rck9_SW1#
```

Verification:**SW1:**

```
Rck9_SW1#ping 192.168.1.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.1.1, timeout is 2 seconds:

```
!!!!!
```

Success rate is 100 percent (5/5), round-trip min/avg/max = 31/31/32 ms

The vlan10 interface is called Switch Virtual Interface (SVI). Each time you create a vlan, the switch try to create an STP (spanning tree protocol) instance and create an SVI interface for that vlan. If the STP instance has been created successfully, then the SVI interface associated with that vlan will be activated.

Task 10:

To accomplish this task use the **ip default-gateway** command. In SW1's global configuration mode, type **ip default-gateway 192.168.1.1**.

SW1:

```
Rck9_SW1#configure terminal
Rck9_SW1(config)#ip default-gateway 192.168.1.1
Rck9_SW1(config)#end
Rck9_SW1#
```

Verification:

To test the configuration, let's display the running-config file and ping R1's loopback0 interface.

SW1:

```
Rck9_SW1#show running-config
Building configuration...
```

```
<omitted output>
```

```
!
```

```
ip default-gateway 192.168.1.1
```

```
!
```

```
<omitted output>
```

```
Rck9_SW1#ping 1.1.1.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:

```
!!!!
```

Success rate is 100 percent (5/5), round-trip min/avg/max = 31/37/47 ms

On Layer 3 switches, the **ip default-gateway** command has no effect when ip routing is enabled. By default ip routing is disabled on Cisco layer-3 switches, so ip default-gateway command will work.

Task 11:

In the interface configuration mode, use the speed command to configure the interface speed. The default value for the speed parameter is auto.

SW1:

```
Rck9_SW1#configure terminal
Rck9_SW1(config)#interface fastethernet 0/1
Rck9_SW1(config-if)#speed 10
Rck9_SW1(config-if)#end
Rck9_SW1#
```

Verification:

SW1:

```
Rck9_SW1#show interfaces fastEthernet 0/1
FastEthernet0/1 is up, line protocol is up (connected)
  Hardware is Fast Ethernet, address is 001f.6d94.7b83 (bia 001f.6d94.7b83)
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 10Mb/s, media type is 10/100BaseTX
<omitted output>
```

Task 12:

In the interface configuration mode use:

- The speed command to configure the interface speed
- The duplex command to configure the interface duplex operation mode.

SW1:

```
Rck9_SW1#configure terminal
Rck9_SW1(config)#interface fastethernet 0/2
Rck9_SW1(config-if)#speed 100
Rck9_SW1(config-if)#duplex full
Rck9_SW1(config-if)#end
Rck9_SW1#
```

The default value of the duplex mode is auto for Fast Ethernet and Gigabit Ethernet ports.

Verification:

```

Rck9_SW1#show interfaces fastethernet 0/2
FastEthernet0/2 is up, line protocol is up (connected)
Hardware is Fast Ethernet, address is 000f.f76d.ac82 (bia 000f.f76d.ac82)
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
  reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
Full-duplex, 100Mb/s, media type is 10/100BaseTX
<omitted output>

```

The duplex operation mode and the speed of the interface are displayed on the same line. On half-duplex links, devices share the bandwidth of the link because each device having to send data should wait until the media is free. This is not the case over full-duplex links, on which devices can send and receive data simultaneously. Therefore, full-duplex mode offers higher throughput than half-duplex mode can offer.

Task 13:

The switch learns MAC addresses either statically or dynamically. Whenever, it receives a frame, it add its source MAC address along with the port on which it was received to table called the CAM table.

Use the mac address-table static global configuration command to add static addresses to the MAC address table.

SW1:

```

Rck9_SW1#configure terminal
Rck9_SW1(config)#mac address-table static ccf4.1256.2559 vlan 10 interface FastEthernet0/24
Rck9_SW1(config)#mac address-table static cc0d.ef01.2345 vlan 10 interface FastEthernet0/23
Rck9_SW1(config)#mac address-table static cc94.1256.2559 vlan 10 interface FastEthernet0/22
Rck9_SW1(config)#mac address-table static cc84.1256.2559 vlan 10 interface FastEthernet0/21
Rck9_SW1(config)#mac address-table static cc74.1256.2559 vlan 20 interface FastEthernet0/20
Rck9_SW1(config)#mac address-table static cc64.1256.2559 vlan 20 interface FastEthernet0/20
Rck9_SW1(config)#mac address-table static cc54.1256.2559 vlan 20 interface FastEthernet0/20
Rck9_SW1(config)#mac address-table static cc44.1256.2559 vlan 20 interface FastEthernet0/20
Rck9_SW1(config)#mac address-table static cc34.1256.2559 vlan 30 interface FastEthernet0/19
Rck9_SW1(config)#mac address-table static cc24.1256.2559 vlan 30 interface FastEthernet0/18
Rck9_SW1(config)#mac address-table static cc14.1256.2559 vlan 30 interface FastEthernet0/17
Rck9_SW1(config)#mac address-table static cc04.1256.2559 vlan 30 interface FastEthernet0/18

```

Verification:**SW1:**Rck9_SW1#**show mac address-table**

Mac Address Table

```

-----
Vlan  Mac Address      Type      Ports
----  -
1     0002.4a2a.942e    DYNAMIC   Fa0/20
1     000a.f324.1037    DYNAMIC   Fa0/10
10    000f.23d5.5220    DYNAMIC   Fa0/1
10    cc0d.ef01.2345    STATIC    Fa0/23
10    cc84.1256.2559    STATIC    Fa0/21
10    cc94.1256.2559    STATIC    Fa0/22
10    ccf4.1256.2559    STATIC    Fa0/24
20    cc44.1256.2559    STATIC    Fa0/20
20    cc54.1256.2559    STATIC    Fa0/20
20    cc64.1256.2559    STATIC    Fa0/20
20    cc74.1256.2559    STATIC    Fa0/20
30    cc04.1256.2559    STATIC    Fa0/18
30    cc14.1256.2559    STATIC    Fa0/17
30    cc24.1256.2559    STATIC    Fa0/18
30    cc34.1256.2559    STATIC    Fa0/19

```

If a switch port appears to have multiple MAC addresses attached to it in the CAM table, it means that the port could be connected to a hub, or a switch. Additionally, it could be configured as access port or trunk link.

Note: How does the switch forward frames?

A unicast frame having a known destination MAC address (figures out in the CAM table) will sent over the associated port in the CAM table. If this port is the same as the port on which the frame is received, then the switch will filter the frame.

A unicast frame has an unknown destination MAC address (does not figure out in the CAM table) will be forwarded out of all ports except the one on which the frame was received.

Broadcast and multicast frames are handled like unicast frames having unknown destination MAC address.

Task 14:

To accomplish this task, use the description command in the interface configuration mode.

SW1:

```
Rck9_SW1#configure terminal
Rck9_SW1(config)#interface fastEthernet0/1
Rck9_SW1(config-if)#description Connected To R1
Rck9_SW1(config-if)#interface fastEthernet0/10
Rck9_SW1(config-if)#description Connected To PC 1
Rck9_SW1(config-if)#interface fastEthernet0/20
Rck9_SW1(config-if)#description Connected To PC 2
Rck9_SW1(config-if)#end
Rck9_SW1#
```

Verification:**SW1:**

```
Rck9_SW1#show interfaces fastEthernet 0/1
FastEthernet0/1 is up, line protocol is up (connected)
Hardware is Fast Ethernet, address is 001f.6d94.7b83 (bia 001f.6d94.7b83)
Description: Connected To R1
```

```
Rck9_SW1#show interfaces fastEthernet 0/10
FastEthernet0/10 is up, line protocol is up (connected)
Hardware is Fast Ethernet, address is 000f.f76d.ac8a (bia 000f.f76d.ac8a)
Description: Connected To PC 1
<omitted output>
```

```
Rck9_SW1#show interfaces fastEthernet 0/20
FastEthernet0/20 is up, line protocol is up (connected)
Hardware is Fast Ethernet, address is 000f.f76d.ac94 (bia 000f.f76d.ac94)
Description: Connected To PC 2
<omitted output>
```

V. Links:

For more information about the commands used in this lab, use the following links:

1. **duplex**

http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli1.html#wp2233221

2. **interface vlan**

http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli1.html#wp2235472

3. **ip address**

http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli1.html#wp1861936

4. **mac address-table static**

http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli1.html#wp2789851

5. **show interfaces**

http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli2.html#wp3339500

6. **show mac address-table**

http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli2.html#wp1942455

7. **show mac address-table dynamic**

http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli2.html#wp1942981

8. **show vlan**

http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli2.html#wp7730585

9. **speed**

http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli3.html#wp1947308

10. **switchport access**

http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli3.html#wp1947308

11. **vlan (global configuration)**

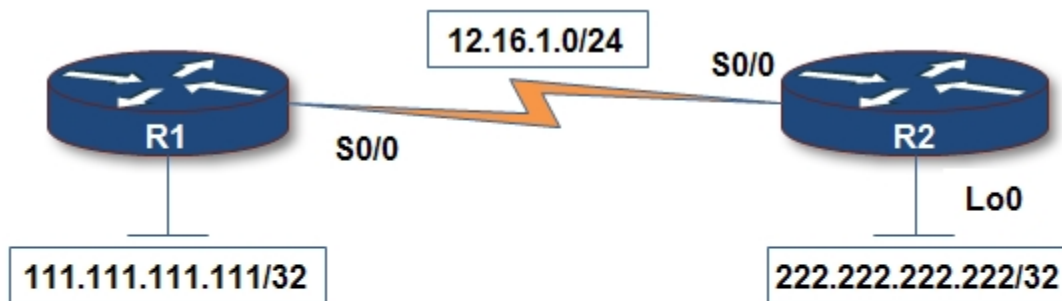
http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli3.html#wp1905882

12. **vlan (VLAN configuration)**

http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli3.html#wp1906101

LAB 1: CDP

I. Network diagram:



CCNA Lab Workbook Volume I v1.0
CDP - Lab 1
© 2009 Mohamed Ouamer
<http://www.networkexperttraining.com>

II. Initial Configuration:

R1:

```
hostname R1
!  
interface Loopback0  
ip address 111.111.111.111 255.255.255.255  
!  
interface Serial0/0  
ip address 12.16.1.1 255.255.255.0  
clockrate 806400  
no shutdown
```

R2:

```
hostname R2
!  
interface Loopback0  
ip address 222.222.222.222 255.255.255.255  
!  
interface Serial0/0  
ip address 12.16.1.2 255.255.255.0  
no shutdown
```

III. Tasks:

1. Display the CDP neighbors of R1 and R2.
2. Configure CDP timers on R1 & R2 as follows

	R1	R2
Timer	5s	10
Holdtime	15s	30s

3. Configure R1 and R2 to insert their loopback0's IP addresses in all CDP packets
4. Disable CDP on the serial0/0 interfaces
5. Disable CDP on the two routers.

IV. Solutions:

Task 1:

R1:

R1#**show cdp neighbors**

Capability Codes: **R** - Router, **T** - Trans Bridge, **B** - Source Route Bridge
S - Switch, **H** - Host, **I** - IGMP, **r** - Repeater

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
R2	Ser 0/0	175	R S I	3640	Ser 0/0

R1#

- The device ID represents the name of the neighbor device. It can also be the MAC address or the serial number of the neighbor device.
- Local Intrfce indicates the interface through the neighbor device is discovered.
- The holdtme field shows the remaining amount of time the current device will wait before it deletes the CDP entry. If the device has received a CDP advertisement for this entry, it will reset the holdtme field.
- The Capability is type of the neighbor device. It tells you what kinds of functions the device has.
- The Platform is product number of the device.
- The Port ID represents the port generating the CDP advertisement on the neighbor device.

R2:

R2#**show cdp neighbors**

Capability Codes: **R** - Router, **T** - Trans Bridge, **B** - Source Route Bridge
S - Switch, **H** - Host, **I** - IGMP, **r** - Repeater

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
R1	Ser 0/0	165	R S I	3640	Ser 0/0

R2#

Task 2:**R1&R2:**

configure terminal
cdp source-interface loopback 0

Verification:**R1:**

R1#**show cdp**
Global CDP information:
Sending CDP packets every 60 seconds
Sending a holdtime value of 180 seconds
Sending CDPv2 advertisements is enabled
Source interface is **Loopback0**

R1#**show cdp neighbors detail | include IP|Device**
Device ID: **R2**
IP address: **222.222.222.222**

R2:

R2#**show cdp**
Global CDP information:
Sending CDP packets every 60 seconds
Sending a holdtime value of 180 seconds
Sending CDPv2 advertisements is enabled
Source interface is **Loopback0**

R2#**show cdp neighbors detail | include IP|Device**
Device ID: **R1**
IP address: **111.111.111.111**

Task 3:**R1:**

```
configure terminal
  cdp timer 5
  cdp holdtime 15
```

R2:

```
configure terminal
  cdp timer 10
  cdp holdtime 30
```

Verification:**R1:**

```
R1#show cdp
```

Global CDP information:

```
  Sending CDP packets every 5 seconds
```

```
  Sending a holdtime value of 15 seconds
```

```
  Sending CDPv2 advertisements is enabled
```

```
R1#debug cdp packets
```

```
CDP packet info debugging is on
```

```
R1#
```

```
*Mar 1 00:30:13.475: CDP-PA: version 2 packet sent out on Serial0/0
```

```
*Mar 1 00:30:18.455: CDP-PA: version 2 packet sent out on Serial0/0
```

```
*Mar 1 00:30:19.691: CDP-PA: Packet received from R2 on interface Serial0/0
```

```
*Mar 1 00:30:19.695: **Entry NOT found in cache**
```

```
*Mar 1 00:30:23.447: CDP-PA: version 2 packet sent out on Serial0/0
```

```
*Mar 1 00:30:28.447: CDP-PA: version 2 packet sent out on Serial0/0
```

```
*Mar 1 00:30:29.723: CDP-PA: Packet received from R2 on interface Serial0/0
```

```
*Mar 1 00:30:29.723: **Entry found in cache**
```

As shown in the output of the debug cdp packets command, the amount of time between two consecutive cdp advertisement sent to R2 is 5 seconds (look at the yellow lines), and the amount of time between two consecutive cdp advertisement received from R2 is 10 seconds (look at the blue lines).

R2:

```
R2#show cdp
```

```
Global CDP information:
```

```
Sending CDP packets every 10 seconds
```

```
Sending a holdtime value of 30 seconds
```

```
Sending CDPv2 advertisements is enabled
```

```
R2#debug cdp packets
```

```
CDP packet info debugging is on
```

```
*Mar 1 00:28:37.031: CDP-PA: Packet received from R1 on interface Serial0/0
```

```
*Mar 1 00:28:37.031: **Entry NOT found in cache**
```

```
*Mar 1 00:28:38.131: CDP-PA: version 2 packet sent out on Serial0/0
```

```
*Mar 1 00:28:41.895: CDP-PA: Packet received from R1 on interface Serial0/0
```

```
*Mar 1 00:28:41.895: **Entry found in cache**
```

```
*Mar 1 00:28:46.891: CDP-PA: Packet received from R1 on interface Serial0/0
```

```
*Mar 1 00:28:46.891: **Entry found in cache**
```

```
*Mar 1 00:28:48.131: CDP-PA: version 2 packet sent out on Serial0/0
```

Task 4:

R1 & R2:

```
configure terminal  
interface serial 0/0  
no cdp enable
```

Verification:

R1:

```
R1#show cdp neighbors
```

```
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
```

S - Switch, H - Host, I - IGMP, r - Repeater

Device ID Local Intrfce Holdtme Capability Platform Port ID

Because CDP is disabled on the serial0/0 interface, router R1 will not discover R2 via this interface as shown in the output of the **show cdp neighbors** command.

R1#**show cdp**

Global CDP information:

Sending CDP packets every 5 seconds
 Sending a holdtime value of 15 seconds
 Sending CDPv2 advertisements is enabled

R1#**show cdp interface**

Serial0/1 is administratively down, line protocol is down

Encapsulation HDLC

Sending CDP packets every 5 seconds

Holdtime is 15 seconds

Serial0/2 is administratively down, line protocol is down

Encapsulation HDLC

Sending CDP packets every 5 seconds

Holdtime is 15 seconds

Serial0/3 is administratively down, line protocol is down

Encapsulation HDLC

Sending CDP packets every 5 seconds

Holdtime is 15 seconds

CDP is still enabled on the router except for the interface serial0/0 as shown in the outputs of the **show cdp** and **show cdp interface** commands.

R2:

R2#**show cdp neighbors**

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge

S - Switch, H - Host, I - IGMP, r - Repeater

Device ID Local Intrfce Holdtme Capability Platform Port ID

R2#**show cdp**

Global CDP information:

Sending CDP packets every 10 seconds

Sending a holdtime value of 30 seconds
Sending CDPv2 advertisements is enabled

R2#show cdp interface

Serial0/1 is administratively down, line protocol is down

Encapsulation HDLC

Sending CDP packets every 10 seconds

Holdtime is 30 seconds

Serial0/2 is administratively down, line protocol is down

Encapsulation HDLC

Sending CDP packets every 10 seconds

Holdtime is 30 seconds

Serial0/3 is administratively down, line protocol is down

Encapsulation HDLC

Sending CDP packets every 10 seconds

Holdtime is 30 seconds

Task 5:**R1 & R2:**

configure terminal
no cdp run

Verification:**R1:**

R1#**show cdp**

% CDP is **not enabled**

R2:

R2#**show cdp**

% CDP is **not enabled**

Now CDP is disabled completely on the two routers.

Links

For more information about the CDP feature, open the following links:

1. cdp advertise-v2
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g01.html#wp1031841
2. cdp enable
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g01.html#wp1031940
3. cdp holdtime
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g01.html#wp1032040
4. cdp log mismatch duplex
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g01.html#wp1068497
5. cdp run
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g01.html#wp1032125
6. cdp source-interface
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g01.html#wp1068876
7. cdp timer
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g01.html#wp1032234
8. clear cdp table
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g01.html#wp1032422
9. Show cdp neighbors
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g07.html#wp1032401
10. show cdp
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g07.html#wp1032872

11. show cdp traffic

http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g07.html#wp1032702

12. show cdp interface

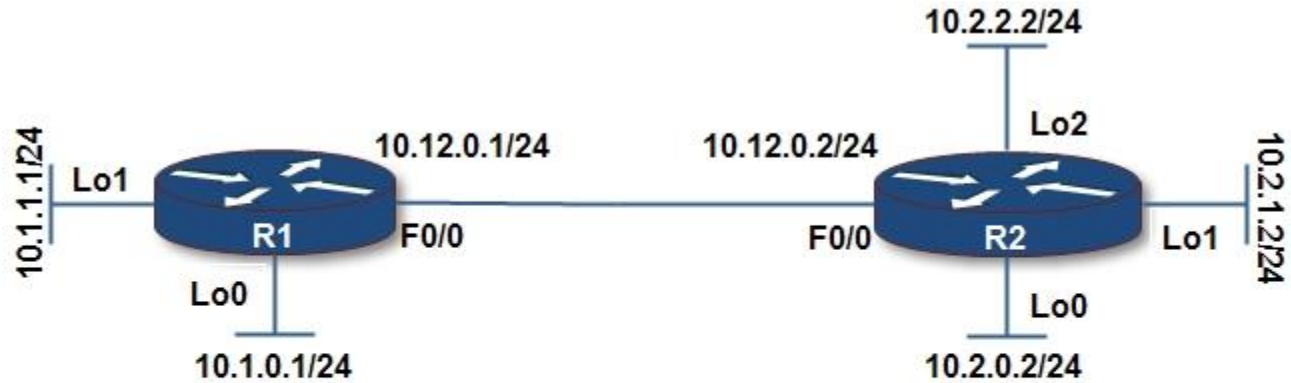
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g07.html#wp1032284

13. show cdp entry

http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g07.html#wp1032137

LAB 2: STATIC IP ROUTING

I. Network diagram:



CCNA Lab Workbook Volume I v1.0
 Static IP Routing - Lab 1
 © 2009 Mohamed Oamer
<http://www.networkexperttraining.com>

II. Initial Configuration:

R1:

```
hostname R1
!
interface loopback0
 ip address 10.1.0.1 255.255.255.0
!
interface loopback1
 ip address 10.1.1.1 255.255.255.0
!
interface fastethernet0/0
 ip address 10.12.0.1 255.255.255.0
 no shutdown
!
!
line console 0
 exec-timeout 0 0
 logging synchronous
!
line vty 0 4
```

```
exec-timeout 0 0
logging synchronous
password cisco
login
```

R2:

```
hostname R2
!
interface loopback0
 ip address 10.2.0.2 255.255.255.0
!
interface loopback1
 ip address 10.2.1.2 255.255.255.0
!
interface loopback2
 ip address 10.2.2.2 255.255.255.0
!
interface fastethernet0/0
 ip address 10.12.0.2 255.255.255.0
 no shutdown
!
!
line console 0
 exec-timeout 0 0
 logging synchronous
!
line vty 0 4
 exec-timeout 0 0
 logging synchronous
 password cisco
 login
```

III. Tasks:

1. Configure two static routes on R2 to reach the subnets connected to R1.
2. On R2, ping the loopback interfaces of R1. The ICMP packets should be source from R2's loopback interfaces.
3. Add necessary static routes to the routing table of R1 so it will be able to reach all the subnets connected to R2.

IV. Solutions:

Task 1:

R1 has three connected subnets. One of them is shared with R2. To reach all R1's subnets, R2 needs to add the routing information of the subnets 10.1.1.0/21 and 10.1.0.0/21 to its routing table. Thus, on R2 add the following statements:

```
ip route 10.1.0.0 255.255.255.0 10.12.0.1
ip route 10.1.1.0 255.255.255.0 10.12.0.1
```

Verification:

R2:

R2#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

10.0.0.0/24 is subnetted, 6 subnets

C 10.12.0.0 is directly connected, FastEthernet0/0

C 10.2.0.0 is directly connected, Loopback0

C 10.2.1.0 is directly connected, Loopback1

C 10.2.2.0 is directly connected, Loopback2

S 10.1.1.0 [1/0] via 10.12.0.1

S 10.1.0.0 [1/0] via 10.12.0.1

Static routes have the code "S" which means static. The default administrative distance is 1 for static routes, and the metric cannot be changed and it is set to 0.

R2#ping 10.1.0.1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.0.1, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 72/106/136 ms

R2#ping 10.1.1.1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 48/96/164 ms

The two pings succeed because they are sourced by the IP address of the Interface f0/0, which is 10.12.0.2, and R1 can route ICMP echo reply back to R2 using the connected route shown in the output of the **show ip route** command above.

You can configure in following manner:

```
ip route 10.1.0.0 255.255.255.0 f0/0
ip route 10.1.1.0 255.255.255.0 f0/0
```

This route will work because the proxy ARP is enabled on R1's f0/0 interface as shown in the output of the **show ip interface f0/0** command below. For each IP packet being forwarded using one of these routes, R2 will generate an ARP request for the IP packet's destination IP address. R1 will respond by inserting the MAC address of its interface F0/0 in the ARP response. At this step, R2 will create an entry in the ARP using the information in the ARP response, and forward the IP packet to R1. The Frame carrying this IP packet has as destination Layer-2 address the MAC address of R1's F0/0.

R1**R1#show ip interface f0/0**

FastEthernet0/0 is up, line protocol is up

Internet address is 10.12.0.1/24

Broadcast address is 255.255.255.255

Address determined by setup command

MTU is 1500 bytes

Helper address is not set

Directed broadcast forwarding is disabled

Outgoing access list is not set

Inbound access list is not set

Proxy ARP is enabled

<omitted output>

The command **ip route subnet subnet-mask interface** will work if the interface parameter represents a point-to-point interface. A point-to-point interface can be any PPP-enabled interface, HDLC-enabled interface, or frame relay point-to-point sub-interface.

In the case of an Ethernet interface, Proxy ARP feature should be enabled on the Forwarding router's interface. The command **ip route subnet subnet-mask interface** will not work if the interface parameter is a main frame relay interface, or a frame relay multipoint sub-interface.

To sum up, try to specify the next hop address as an IP address instead of an exit interface whenever you have to configure a static route.

Task 2:

Before trying to ping R1's loopback interfaces, enable IP packet debugging to see why R2 will not get any ICMP echo-reply packet from R1.

R2:

```
R2#ping 10.1.0.1 source loopback 0
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.0.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.0.2

.....

Success rate is 0 percent (0/5)

```
R2#ping 10.1.0.1 source loopback 1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.0.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.1.2

.....

Success rate is 0 percent (0/5)

```
R2#ping 10.1.0.1 source loopback 2
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.0.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.2.2

.....

Success rate is 0 percent (0/5)

R2#ping 10.1.1.1 source loopback 0

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.0.2

.....

Success rate is 0 percent (0/5)

R2#ping 10.1.1.1 source loopback 1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.1.2

.....

Success rate is 0 percent (0/5)

R2#ping 10.1.1.1 source loopback 2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.2.2

.....

Success rate is 0 percent (0/5)

R2#

As shown in the output of the **show ip route** command below, the routing table of R1 has three routes. Any of these routes cannot be used to route IP packets with a destination of 10.2.0.2, 10.2.1.2, or 10.2.2.2. This is why the pings did not succeed. Additionally, the debug IP packet output shows that IP packet with a destination IP addresses of 10.2.0.2, 10.2.1.2, or 10.2.2.2 are unroutable.

R1#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

10.0.0.0/24 is subnetted, 3 subnets

C 10.12.0.0 is directly connected, FastEthernet0/0

C 10.1.1.0 is directly connected, Loopback1

C 10.1.0.0 is directly connected, Loopback0

R1

R1#**debug ip packet**

IP packet debugging is on

R1#

IP: tableid=0, s=10.2.0.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.0.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.0.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.0.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.0.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.0.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.0.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.0.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.0.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.0.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.0.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.0.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.0.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.0.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.0.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.1.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.1.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.1.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.1.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.1.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.1.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.1.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.1.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.1.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.1.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.1.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.1.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.1.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.1.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.1.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.2.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.2.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.2.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.2.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.2.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.2.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.2.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.2.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.2.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.2.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.2.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.2.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.2.2 (FastEthernet0/0), d=10.1.0.1 (Loopback0), routed via RIB

IP: s=10.2.2.2 (FastEthernet0/0), d=10.1.0.1, len 100, rcvd 4

IP: s=10.1.0.1 (local), **d=10.2.2.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.0.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB

IP: s=10.2.0.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4

IP: s=10.1.1.1 (local), **d=10.2.0.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.0.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB

IP: s=10.2.0.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4

IP: s=10.1.1.1 (local), **d=10.2.0.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.0.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB

IP: s=10.2.0.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4

IP: s=10.1.1.1 (local), **d=10.2.0.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.0.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB

IP: s=10.2.0.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4

IP: s=10.1.1.1 (local), **d=10.2.0.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.0.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB

IP: s=10.2.0.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4

IP: s=10.1.1.1 (local), **d=10.2.0.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.1.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB

IP: s=10.2.1.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4

IP: s=10.1.1.1 (local), **d=10.2.1.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.1.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB

IP: s=10.2.1.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4

IP: s=10.1.1.1 (local), **d=10.2.1.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.1.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB
 IP: s=10.2.1.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4
 IP: s=10.1.1.1 (local), **d=10.2.1.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.1.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB
 IP: s=10.2.1.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4
 IP: s=10.1.1.1 (local), **d=10.2.1.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.1.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB
 IP: s=10.2.1.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4
 IP: s=10.1.1.1 (local), **d=10.2.1.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.2.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB
 IP: s=10.2.2.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4
 IP: s=10.1.1.1 (local), **d=10.2.2.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.2.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB
 IP: s=10.2.2.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4
 IP: s=10.1.1.1 (local), **d=10.2.2.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.2.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB
 IP: s=10.2.2.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4
 IP: s=10.1.1.1 (local), **d=10.2.2.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.2.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB
 IP: s=10.2.2.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4
 IP: s=10.1.1.1 (local), **d=10.2.2.2**, len 100, **unroutable**

R1#

IP: tableid=0, s=10.2.2.2 (FastEthernet0/0), d=10.1.1.1 (Loopback1), routed via RIB
 IP: s=10.2.2.2 (FastEthernet0/0), d=10.1.1.1, len 100, rcvd 4
 IP: s=10.1.1.1 (local), **d=10.2.2.2**, len 100, **unroutable**

Task 3:

To accomplish this task, you can follow different ways. You can configure a static route for each subnet connected to R2, except for the one shared with R1. You can also configure a default route on R1 that points to R2.

R2:

1st method:

```
ip route 10.2.0.0 255.255.255.0 f0/0
ip route 10.2.1.0 255.255.255.0 f0/0
ip route 10.2.2.0 255.255.255.0 f0/0
```

Verification:

R1#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

10.0.0.0/24 is subnetted, 6 subnets
C 10.12.0.0 is directly connected, FastEthernet0/0
S 10.2.0.0 is directly connected, FastEthernet0/0
S 10.2.1.0 is directly connected, FastEthernet0/0
S 10.2.2.0 is directly connected, FastEthernet0/0
C 10.1.1.0 is directly connected, Loopback1
C 10.1.0.0 is directly connected, Loopback0

R1#ping 10.2.0.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.0.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 72/128/220 ms

R1#ping 10.2.1.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.1.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 20/109/164 ms

R1#ping 10.2.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
.!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/292/572 ms
R1#

2nd method:

```
ip route 0.0.0.0 0.0.0.0 10.12.0.2
```

Verification:

```
R1#show ip route
<omitted output>
```

```
Gateway of last resort is 10.12.0.2 to network 0.0.0.0
```

```
10.0.0.0/24 is subnetted, 3 subnets
C    10.12.0.0 is directly connected, FastEthernet0/0
C    10.1.1.0 is directly connected, Loopback1
C    10.1.0.0 is directly connected, Loopback0
S*  0.0.0.0/0 [1/0] via 10.12.0.2
```

The line “**Gateway of last resort is 10.12.0.2 to network 0.0.0.0**” shows that the default gateway of the router is 10.12.0.2. If you configure the static route using the **ip route 0.0.0.0 0.0.0.0 f0/0** command, R1 will show you the message “Gateway of last resort is **0.0.0.0** to network 0.0.0.0”. If the router does not have a gateway of last resort, the **show ip route** command will display the message “Gateway of last resort is not set”.

Now, let’s ping R2’s loopback interfaces.

R1#ping 10.2.0.2

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.0.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 72/128/220 ms
```

R1#ping 10.2.1.2

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.1.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 20/109/164 ms
```

R1#ping 10.2.2.2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:

.!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 28/292/572 ms

R1#

R2:**R2#ping 10.1.0.1 source loopback 0**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.0.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.0.2

.!!!!

Success rate is 80 percent (4/5), round-trip min/avg/max = 44/118/156 ms

R2#ping 10.1.0.1 source loopback 1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.0.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.1.2

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 52/120/232 ms

R2#ping 10.1.0.1 source loopback 2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.0.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.2.2

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 44/108/144 ms

R2#ping 10.1.1.1 source loopback 0

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.0.2

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 96/120/144 ms

R2#ping 10.1.1.1 source loopback 1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.1.2

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 44/147/240 ms

R2#ping 10.1.1.1 source loopback 2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

Packet sent with a source address of 10.2.2.2

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 28/113/240 ms

R2#

When R1 receives an ICMP echo-request packet from R2, it will look in its routing table to decide how to forward the associated ICMP echo-reply packet back to R2. Because R1 has full routing information about R2's connected subnets, it can reply to any ICMP echo-request packet generated from any subnet connected to R2.

By default, static IP routes have an administrative distance of 1 until otherwise configured. If the route points to an interface, it will be shown as a connected route but the administrative distance still has a value of 1.

V. Links

For more information about the commands used in this lab, use the following links:

1. **debug ip packet**
http://www.cisco.com/en/US/docs/ios/12_3/debug/command/reference/dbg_i2g.html#wp1086651
2. **ip route**
http://www.cisco.com/en/US/docs/ios/12_3/iproute/command/reference/ip2_i1g.html#wp1040624
3. **ping**
http://www.cisco.com/en/US/docs/ios/12_3/configfun/command/reference/cfr_1g06.html#wp1075989
4. **show ip interface**
http://www.cisco.com/en/US/docs/ios/12_3/interface/command/reference/int_s3g.html#wp1205362
5. **show ip route**
http://www.cisco.com/en/US/docs/ios/12_3/iproute/command/reference/ip2_s3g.html#wp1039689

LAB 3: PORT SECURITY

I. Network diagram:



CCNA Lab Workbook Volume I v1.0
Port Security - Lab 1
© 2009 Mohamed Ouamer
<http://www.networkexperttraining.com>

II. Initial Configuration:

R1:

```
hostname R1
!  
interface f0/0  
ip address 10.0.0.1 255.0.0.0  
no shut  
!  
!  
line console 0  
exec-timeout 0 0  
logging synchronous  
!  
line vty 0 4  
exec-timeout 0 0  
logging synchronous  
password cisco  
login
```

SW1:

```
hostname SW1
!  
interface vlan1  
ip address 10.0.0.2 255.0.0.0  
!  
!  
line console 0  
exec-timeout 0 0  
logging synchronous  
!  
line vty 0 4  
exec-timeout 0 0  
logging synchronous  
password cisco  
login
```

III. Tasks:

4. Configure the switch to permit frames sourced by R1's F0/0 interface.
5. Set the maximum number of MAC addresses that will be allowed on switch port f0/1 to the value of 2.

IV. Solutions:

Task 1:

To accomplish this task you should follow these steps:

1. Retrieve the MAC address on R1's F0/0 interface by using the command **show interface F0/0**.

```
R1#show interfaces f0/0
FastEthernet0/0 is up, line protocol is up
  Hardware is AmdFE, address is cc00.0ebc.0000 (bia cc00.0ebc.0000)
```

The MAC address of R1's F0/0 interface is **cc00.0ebc.0000** as shown in the output of the command show interfaces f0/0.

The bia keyword means burned in address. The bia here is the same as the MAC address used to source frames out of the interface F0/0. You cannot modify the bia MAC address, but the Cisco IOS allows you to modify the MAC address used to source frames out a specific interface.

Another way to retrieve the MAC address of R1's F0/0 interface is to use the show mac address-table.

```
SW1#show mac address-table
  <<Omitted output>>
  1 cc00.0ebc.0000 Dynamic FastEthernet0/1
```

The MAC address **cc00.0ebc.0000** is learned dynamically over the interface F0/1. Because one device (the router R1) is attached to the SW1's F0/1 port, this Mac is the MAC address of the interface F0/0 on R1.

Note: In our case, we can make SW1 learn the MAC address **cc00.0ebc.0000** via CDP or by sending an ICMP echo-echo packet to its vlan1 interface from R1.

2. Set the mode of the port F0/1 to access using the command **switchport mode access**.
3. Enable the port-security feature on SW1's F0/1 interface by using the command **switchport port-security**.
4. Configure the MAC address of R1's F0/0 interface, as secure MAC on the port F0/0. Use the command **switchport port-security mac-address**.
5. Save the configuration.

So SW1 will be configured as follows:

```
Interface f0/1
  Switchport mode access
  Switchport port-security
  Switchport port-security mac-address cc00.0ebc.0000
End
```

Copy running-config startup-config

Verification:

To verify our configuration we'll use the commands below on SW1:

- Show running-config interface f0/1
SW1# **show running-config interface f0/1**
Interface f0/1
 Switchport mode access
 Switchport port-security
 Switchport port-security mac-address cc00.0ebc.0000
- show port-security interface fastethernet 0/1

```
SW1#show port-security interface fastethernet 0/1
Port Security : Enabled
Port Status : Secure-up
Violation Mode : Shutdown
Aging Time : 0 mins
Aging Type : Absolute
SecureStatic Address Aging : Disabled
```

Maximum MAC Addresses : 1

Total MAC Addresses : 1

Configured MAC Addresses : 1

Sticky MAC Addresses : 0

Last Source Address:Vlan : **cc00.0ebc.0000:1**

Security Violation Count : 0

The line "**Port Security : Enabled**" means that the port security feature is enabled on the interface f0/1.

The line "**Port Status : Secure-up**" means that the port is up and there is no violation on the interface f0/1.

The line "**Configured MAC Addresses : 1**" means that the running configuration of the interface f0/1 contains one "switchport port-security mac-address" statement.

The line "**Last Source Address:Vlan : cc00.0ebc.0000:1**" shows that the MAC address **cc00.0ebc.0000** belongs to a device in vlan 1 and is the last MAC address learned on the interface f0/1.

The line "**Security Violation Count : 0**" shows that there is no violation on the interface f0/1.

By default the violation mode is set to "shutdown", which means that the port will be put in the shutdown state if the switch has learned a number of MAC addresses more than the maximum allowed.

To verify this we'll instruct the router to change the MAC address used as the source address field in the frames exiting its interface F0/0.

On R1, enter the following configuration:

```
Interface f0/0
  mac-address 1234.5678.9abc
```

```
R1#show interfaces f0/0
FastEthernet0/0 is up, line protocol is up
Hardware is AmdFE, address is 1234.5678.9abc (bia cc00. 0ebc.0000)
```

On SW1, use these commands to see what happened:

```
SW1#show interface fa0/1
FastEthernet0/1 is down, line protocol is down (err-disabled)
<<Omitted output>>
```

```
SW1#show port-security interface fa0/1
Port Security : Enabled
Port Status : Secure-shutdown
Violation Mode : Shutdown
Aging Time : 0 mins
Aging Type : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses : 1
Total MAC Addresses : 1
Configured MAC Addresses : 1
Sticky MAC Addresses : 0
Last Source Address:Vlan : 1234.5678.9abc:1
Security Violation Count : 1
```

After receiving the first frame source by the MAC address **1234.5678.9abc** on the port F0/1., SW1 has putted the port F0/1 into err-disable state.

The output of the command **show interface fa0/1** shows that the interface **F0/1** is down.

The output of the command **show port-security interface fa0/1** shows that the port status is secure-shutdown witch means that a violation has occurred.

To reenale the interface f0/1 we have either to use the no shutdown command, or configure the errdisable recovery command.

Task 2:

This task can be done using the command `switchport port-security maximum`.

```
Interface fastethernet 0/1
  switchport port-security maximum 2
```

To verify our configuration we'll use the commands below on SW1:

- Show running-config interface f0/1
SW1# **show running-config interface f0/1**
Interface f0/1
 <<Omitted output>>
 switchport port-security maximum 2

- show port-security interface fastethernet 0/1

```
SW1#show port-security interface fastethernet 0/1
Port Security : Enabled
Port Status : Secure-up
Violation Mode : Shutdown
Aging Time : 0 mins
Aging Type : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses : 2
Total MAC Addresses : 1
Configured MAC Addresses : 1
Sticky MAC Addresses : 0
Last Source Address:Vlan : cc00.0ebc.0000:1
Security Violation Count : 0
```

The output of the `show port-security interface fastethernet 0/1` command shows that the maximum number of the secure Mac addresses allowed on the port f0/01 is 2.

V. Links

For more information about the Switch port security feature, use the following links:

1. Configuring Port Security
http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/configuration/guide/swtrafc.html#wp1038501
2. Understanding Port Security
http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/configuration/guide/swtrafc.html#wp1042596
3. Default Port Security Configuration
http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/configuration/guide/swtrafc.html#wp1038518
4. Port Security Configuration Guidelines
http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/configuration/guide/swtrafc.html#wp1038546
5. Enabling and Configuring Port Security
http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/configuration/guide/swtrafc.html#wp1155336
6. show port-security command:
http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli2.html#wp7628580
7. Switchport port-security
http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_44se/command/reference/cli3.html#wp1948361