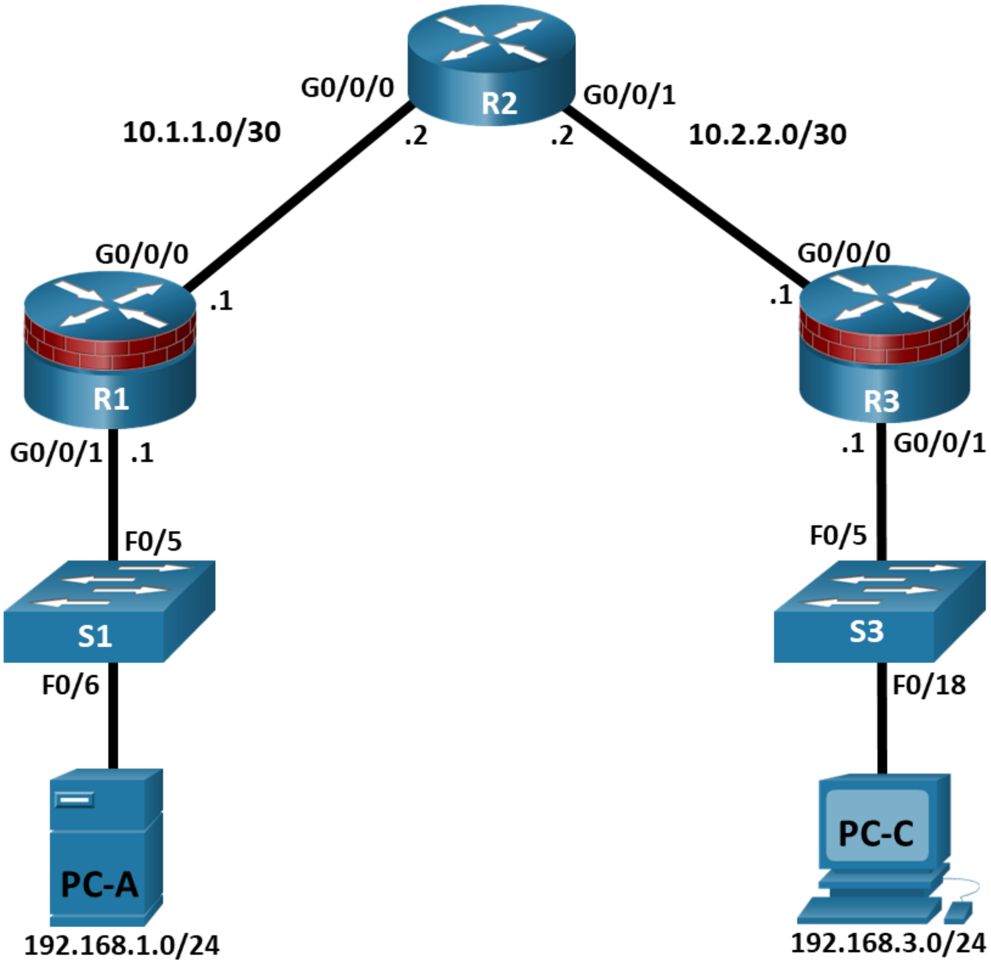
# Lab - Configure Cisco IOS Resilience Management and Reporting

## Topology



## Addressing Table

| Device | Interface | IP Address | Subnet Mask | Default Gateway | Switch Port |
|---|---|---|---|---|---|
| R1 | G0/0/0 | 10.1.1.1 | 255.255.255.252 | N/A | N/A |
| | G0/0/1 | 192.168.1.1 | 255.255.255.0 | N/A | S1 F0/5 |
| R2 | G0/0/0 | 10.1.1.2 | 255.255.255.252 | N/A | N/A |
| | G0/0/1 | 10.2.2.2 | 255.255.255.252 | N/A | N/A |
| R3 | G0/0/0 | 10.2.2.1 | 255.255.255.252 | N/A | N/A |
| | G0/0/1 | 192.168.3.1 | 255.255.255.0 | N/A | S3 F0/5 |

| Device | Interface | IP Address | Subnet Mask | Default Gateway | Switch Port |
|---|---|---|---|---|---|
| PC-A | NIC | 192.168.1.3 | 255.255.255.0 | 192.168.1.1 | S1 F0/6 |
| PC-C | NIC | 192.168.3.3 | 255.255.255.0 | 192.168.3.1 | S3 F0/18 |

## Objectives

**Part 1: Configure Basic Device Settings**

**Part 2: Configure SNMPv3 Security using an ACL.**

**Part 3: Configure a router as a synchronized time source for other devices using NTP.**

**Part 4: Configure syslog support on a router.**

## Background / Scenario

The router is a critical component in any network. It controls the movement of data into and out of the network and between devices within the network. It is particularly important to protect network routers because the failure of a routing device could make sections of the network, or the entire network, inaccessible. Controlling access to routers and enabling reporting on routers is critical to network security and should be part of a comprehensive security policy.

In this lab, you will build a multi-router network and configure the routers and hosts. You will configure SNMP, NTP, and syslog support to monitor router configuration changes.

**Note**: The routers used with hands-on labs are Cisco 4221 with Cisco IOS XE Release 16.9.6 (universalk9 image). The switches used in the labs are Cisco Catalyst 2960+ with Cisco IOS Release 15.2(7) (lanbasek9 image). Other routers, switches, and Cisco IOS versions can be used. Depending on the model and Cisco IOS version, the commands available and the output produced might vary from what is shown in the labs. Refer to the Router Interface Summary Table at the end of the lab for the correct interface identifiers.

**Note**: Before you begin, ensure that the routers and the switches have been erased and have no startup configurations.

## Required Resources

- 3 Routers (Cisco 4221 with Cisco XE Release 16.9.6 universal image or comparable with a Security Technology Package license)
- 2 Switches (Cisco 2960+ with Cisco IOS Release 15.2(7) lanbasek9 image or comparable)
- 2 PCs (Windows OS with a terminal emulation program, such as PuTTY or Tera Term installed)
- Console cables to configure Cisco networking devices
- Ethernet cables as shown in the topology

## Instructions

## Part 1: Configure Basic Device Settings

In this part, set up the network topology and configure basic settings, such as interface IP addresses.

### Step 1: Cable the network.

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

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

a. Console into the router and enable privileged EXEC mode.

```
Router> enable
Router# configure terminal
```

b. Configure host names as shown in the topology.

```
R1(config)# hostname R1
```

c. Configure interface IP addresses as shown in the IP Addressing Table.

```
R1(config)# interface g0/0/0
R1(config-if)# ip address 10.1.1.1 255.255.255.0
R1(config-if)# no shutdown

R1(config)# interface g0/0/1
R1(config-if)# ip address 192.168.1.1 255.255.255.0
R1(config-if)# no shutdown
```

d. To prevent the router from attempting to translate incorrectly entered commands as though they were host names, disable DNS lookup. R1 is shown here as an example.

```
R1(config)# no ip domain-lookup
```

### Step 3: Configure OSPF routing on the routers.

a. Use the **router ospf** command in global configuration mode to enable OSPF on R1.

```
R1(config)# router ospf 1
```

b. Configure the **network** statements for the networks on R1. Use an area ID of 0.

```
R1(config-router)# network 192.168.1.0 0.0.0.255 area 0
R1(config-router)# network 10.1.1.0 0.0.0.3 area 0
```

c. Configure OSPF on R2 and R3.

```
R2(config)# router ospf 1
R2(config-router)# network 10.1.1.0 0.0.0.3 area 0
R2(config-router)# network 10.2.2.0 0.0.0.3 area 0

R3(config)# router ospf 1
R3(config-router)# network 10.2.2.0 0.0.0.3 area 0
R3(config-router)# network 192.168.3.0 0.0.0.255 area 0
```

d. Issue the **passive-interface** command to change the G0/0/1 interface on R1 and R3 to passive.

```
R1(config)# router ospf 1
R1(config-router)# passive-interface g0/0/1

R3(config)# router ospf 1
R3(config-router)# passive-interface g0/0/1
```

### Step 4: Verify OSPF neighbors and routing information.

a. Issue the **show ip ospf neighbor** command to verify that each router lists the other routers in the network as neighbors.

```
R1# show ip ospf neighbor
```

```
Neighbor ID     Pri   State           Dead Time   Address         Interface
10.2.2.2          1   FULL/BDR        00:00:37    10.1.1.2        GigabitEthernet0/0/0
```

b. Issue the **show ip route** command to verify that all networks display in the routing table on all routers.

```
R1# show ip route
Codes: L - local, 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, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override, p - overrides from PfR

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
C        10.1.1.0/30 is directly connected, GigabitEthernet0/0/0
L        10.1.1.1/32 is directly connected, GigabitEthernet0/0/0
O        10.2.2.0/30 [110/2] via 10.1.1.2, 00:01:11, GigabitEthernet0/0/0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, GigabitEthernet0/0/1
L        192.168.1.1/32 is directly connected, GigabitEthernet0/0/1
O     192.168.3.0/24 [110/3] via 10.1.1.2, 00:01:07, GigabitEthernet0/0/0
```

### Step 5: Configure PC host IP settings.

Configure a static IP address, subnet mask, and default gateway for PC-A and PC-C as shown in the IP Addressing Table.

### Step 6: Verify connectivity between PC-A and PC-C.

a. Ping from R1 to R3.

   If the pings are not successful, troubleshoot the basic device configurations before continuing.

b. Ping from PC-A, on the R1 LAN, to PC-C, on the R3 LAN.

   If the pings are not successful, troubleshoot the basic device configurations before continuing.

**Note**: If you can ping from PC-A to PC-C you have demonstrated that OSPF routing is configured and functioning correctly. If you cannot ping but the device interfaces are up and IP addresses are correct, use the **show run, show ip ospf neighbor,** and **show ip route** commands to help identify routing protocol-related problems.

### Step 7: Save the basic running configuration for each router.

Save the basic running configuration for the routers as text files on your PC. These text files can be used to restore configurations later in the lab.

## Part 2: Configure SNMPv3 Security using an ACL.

Simple Network Management Protocol (SNMP) enables network administrators to monitor network performance, mange network devices, and troubleshoot network problems. SNMPv3 provides secure access by authenticating and encrypting SNMP management packets over the network. You will configure SNMPv3 using an ACL on R1.

### Step 1: Configure an ACL on R1 that will restrict access to SNMP on the 192.168.1.0 LAN.

a. Create a standard access-list named **PERMIT-SNMP**.

```
R1(config)# ip access-list standard PERMIT-SNMP
```

b. Add a permit statement to allow only packets on R1's LAN.

```
R1(config-std-nacl)# permit 192.168.1.0 0.0.0.255
R1(config-std-nacl)# exit
```

### Step 2: Configure the SNMP view.

Configure a SNMP view called **SNMP-RO** to include the ISO MIB family.

```
R1(config)# snmp-server view SNMP-RO iso included
```

### Step 3: Configure the SNMP group.

Call the group name **SNMP-G1**, and configure the group to use SNMPv3 and require both authentication and encryption by using the **priv** keyword. Associate the view you created in Step 2 to the group, giving it read only access with the **read** parameter. Finally specify the ACL **PERMIT-SNMP**, configured in Step 1, to restrict SNMP access to the local LAN.

```
R1(config)# snmp-server group SNMP-G1 v3 priv read SNMP-RO access PERMIT-SNMP
```

### Step 4: Configure the SNMP user.

Configure an **SNMP-Admin** user and associate the user to the **SNMP-G1** group you configured in Step 3. Set the authentication method to **SHA** and the authentication password to **Authpass**. Use AES-128 for encryption with a password of **Encrypass**.

```
R1(config)# snmp-server user SNMP-Admin SNMP-G1 v3 auth sha Authpass priv aes
128 Encrypass
R1(config)# end
```

### Step 5: Verify your SNMP configuration.

a. Use the **show snmp group** command in privilege EXEC mode to view the SNMP group configuration. Verify that your group is configured correctly.

**Note**: If you need to make changes to the group, use the command **no snmp group** to remove the group from the configuration and then re-add it with the correct parameters.

```
R1# show snmp group
groupname: ILMI                             security model:v1
contextname: <no context specified>         storage-type: permanent
readview : *ilmi                            writeview: *ilmi
notifyview: <no notifyview specified>
row status: active

groupname: ILMI                             security model:v2c
contextname: <no context specified>         storage-type: permanent
readview : *ilmi                            writeview: *ilmi
notifyview: <no notifyview specified>
row status: active
```

```
groupname: SNMP-G1                          security model:v3 priv
contextname: <no context specified>         storage-type: nonvolatile
readview : SNMP-RO                          writeview: <no writeview specified>
notifyview: <no notifyview specified>
row status: active      access-list: PERMIT-SNMP
```

b. Use the command **show snmp user** to view the SNMP user information.

**Note**: The **snmp-server user** command is hidden from view in the configuration for security reasons. However, if you need to make changes to a SNMP user, you can issue the command **no snmp-server user** to remove the user from the configuration, and then re-add the user with the new parameters.

```
R1# show snmp user

User name: SNMP-Admin
Engine ID: 800000090300707 9B3923640
storage-type: nonvolatile        active
Authentication Protocol: SHA
Privacy Protocol: AES128
Group-name: SNMP-G1
```

## Part 3: Configure a Synchronized Time Source Using NTP.

R2 will be the master NTP clock source for routers R1 and R3.

**Note**: R2 could also be the master clock source for switches S1 and S3, but it is not necessary to configure them for this lab.

### Step 1: Set Up the NTP Master using Cisco IOS commands.

R2 is the master NTP server in this lab. All other routers and switches learn the time from it, either directly or indirectly. For this reason, you must ensure that R2 has the correct Coordinated Universal Time set.

a. Use the **show clock** command to display the current time set on the router.

```
R2# show clock
*18:18:25.443 UTC Sun Jan 31 2021
```

b. To set the time on the router, use the **clock set** *time* command.

```
R2# clock set 11:17:00 Jan 31 2021
R2#
*Jan 31 11:17:00.001: %SYS-6-CLOCKUPDATE: System clock has been updated from
18:19:03 UTC Sun Jan 31 2021 to 11:17:00 UTC Sun Jan 31 2021, configured from
console by console.
Jan 31 11:17:00.001: %PKI-6-AUTHORITATIVE_CLOCK: The system clock has been
set.
```

c. Configure NTP authentication by defining the authentication key number, hashing type, and password that will be used for authentication. The password is case sensitive.

```
R2# config t
R2(config)# ntp authentication-key 1 md5 NTPpassword
```

d. Configure the trusted key that will be used for authentication on R2.

```
R2(config)# ntp trusted-key 1
```

e. Enable the NTP authentication feature on R2.

```
R2(config)# ntp authenticate
```

f. Configure R2 as the NTP master using the **ntp master** *stratum-number* command in global configuration mode. The stratum number indicates the distance from the original source. For this lab, use a stratum number of **3** on R2. When a device learns the time from an NTP source, its stratum number becomes one greater than the stratum number of its source.

```
R2(config)# ntp master 3
```

### Step 2: Configure R1 and R3 as NTP clients using the CLI.

a. Issue the **debug ntp all** command to see NTP activity on R1 as it synchronizes with R2. Review the debug messages as you proceed through this step.

```
R1# debug ntp all
NTP events debugging is on
NTP core messages debugging is on
NTP clock adjustments debugging is on
NTP reference clocks debugging is on
NTP packets debugging is on
```

b. Configure NTP authentication by defining the authentication key number, hashing type, and password that will be used for authentication.

```
R1# config t
R1(config)# ntp authentication-key 1 md5 NTPpassword
R1(config)#
*Jan 31 18:41:23.707: NTP Core(INFO): keys initilized.
*Jan 31 18:41:23.712: NTP Core(NOTICE): proto: precision =  usec
*Jan 31 18:41:23.712: %NTP : Drift Read Failed (String Error).
*Jan 31 18:41:23.712: NTP Core(DEBUG): drift value read: 0.000000000
*Jan 31 18:41:23.712: NTP Core(NOTICE): ntpd  PPM
*Jan 31 18:41:23.712: NTP Core(NOTICE): trans state : 1
*Jan 31 18:41:23.712: NTP: Initialized interface GigabitEthernet0/0/0
*Jan 31 18:41:23.712: NTP: Initialized interface GigabitEthernet0/0/1
*Jan 31 18:41:23.712: NTP: Initialized interface LIIN0
R1(config)#
*Jan 31 18:41:23.713: NTP Core(INFO): more memory added for keys.
*Jan 31 18:41:23.713: NTP Core(INFO): key (1) added.
```

c. Configure the trusted key that will be used for authentication. This command provides protection against accidentally synchronizing the device to a time source that is not trusted.

```
R1(config)# ntp trusted-key 1
R1(config)#
*Jan 31 18:43:56.191: NTP Core(INFO): key (1) marked as trusted.
```

d. Enable the NTP authentication feature.

```
R1(config)# ntp authenticate
R1(config)#
*Jan 31 18:44:33.482: NTP Core(INFO): 0.0.0.0 C01C 0C clock_step
```

e. R1 and R3 will become NTP clients of R2. Use the command **ntp server** *hostname*. The host name can also be an IP address.

**Note**: The command **ntp update-calendar** may be necessary to periodically updates the calendar with the NTP time for other IOS images.

```
R1(config)# ntp server 10.1.1.2
R1(config)#
*Jan 31 18:45:29.714: NTP message sent to 10.1.1.2, from interface
'GigabitEthernet0/0/0' (10.2.2.1).
*Jan 31 18:45:29.715: NTP message received from 10.1.1.2 on interface
'GigabitEthernet0/0/0' (10.2.2.1).
*Jan 31 18:45:29.716: NTP Core(DEBUG): ntp_receive: message received
*Jan 31 18:45:29.716: NTP Core(DEBUG): ntp_receive: peer is 0x80007FA135BB32F8, next
action is 1.
*Jan 31 18:45:29.716: NTP Core(DEBUG): Peer becomes reachable, poll set to 6.

*Jan 31 18:45:29.716: NTP Core(INFO): 10.1.1.2 8014 84 reachable
*Jan 31 18:45:29.716: NTP Core(INFO): 10.1.1.2 962A 8A sys_peer
R1(config)#
*Jan 31 18:45:29.716: NTP: step(0xFFFF9D56.B5A1C9F4): local_offset =
0x00000000.00000000, curtime = 0xE3C17949.B74BC8A0
*Jan 31 11:44:32.426: NTP Core(NOTICE): time reset -25257.290500 s
*Jan 31 11:44:32.426: NTP Core(NOTICE): trans state : 4
*Jan 31 11:44:32.426: NTP Core(INFO): 0.0.0.0 C62C 0C clock_step
*Jan 31 11:44:32.426: NTP Core(INFO): 0.0.0.0 C03C 0C clock_step
*Jan 31 11:44:33.423: NTP message sent to 10.1.1.2, from interface
'GigabitEthernet0/0/0' (10.2.2.1).
*Jan 31 11:44:33.424: NTP message received from 10.1.1.2 on interface
'GigabitEthernet0/0/0' (10.2.2.1).
*Jan 31 11:44:33.424: NTP Core(DEBUG): ntp_receive: message received
*Jan 31 11:44:33.424: NTP Core(DEBUG): ntp_receive: peer is 0x80007FA135BB32F8, next
action is 1.
*Jan 31 11:44:33.424: NTP Core(DEBUG): Peer becomes reachable, poll set to 6.

*Jan 31 11:44:33.424: NTP Core(INFO): 10.1.1.2 8034 84 reachable
*Jan 31 11:44:33.425: NTP Core(INFO): 10.1.1.2 964A 8A sys_peer
```

f. Issue the **undebug all** or the **no debug ntp all** command to turn off debugging.

```
R1# undebug all
```

g. Verify that R1 has made an association with R2 with the **show ntp associations** command. You can also use the more verbose version of the command by adding the **detail** argument. It might take some time for the NTP association to form.

```
R1# show ntp associations

address    ref clock    st  when  poll reach  delay  offset   disp
~10.1.1.2  127.127.1.1   3    14    64     3  0.000  -280073  3939.7
*sys.peer, # selected, +candidate, -outlyer, x falseticker, ~ configured
```

h. Verify the time on R1 after it has made an association with R2.

```
R1# show clock
*11:49:27.709 UTC Sun Jan 31 2021
```

i. Repeat the NTP configurations to configure R3 as an NTP client.

## Part 4: Configure syslog Support on R1 and PC-A.

### Step 1: Install and start the syslog server.

Free or trial versions of syslog server can be downloaded from the Internet. Use a web browser to search for “free windows syslog server” and refer to the software documentation for more information. Your instructor may also recommend a suitable syslog server for classroom use.

If a syslog server is not currently installed on the host, download a syslog server and install it on PC-A. If it is already installed, go to the next step.

### Step 2: Configure R1 to log messages to the syslog server using the CLI.

a. Start the syslog server.

b. Verify that you have connectivity between R1 and PC-A by pinging the R1 G0/0/1 interface IP address 192.168.1.1. If it is not successful, troubleshoot as necessary before continuing.

c. NTP was configured in a previous part to synchronize the time on the network. Displaying the correct time and date in syslog messages is vital when using syslog to monitor a network. If the correct time and date of a message is not known, it can be difficult to determine what network event caused the message.

   Verify that the timestamp service for logging is enabled on the router using the **show run** command. Use the following command if the timestamp service is not enabled.

```
R1(config)# service timestamps log datetime msec
```

d. Configure the syslog service on the router to send syslog messages to the syslog server.

```
R1(config)# logging host 192.168.1.3
```

### Step 3: Configure the logging severity level on R1.

Logging traps can be set to support the logging function. A trap is a threshold that when reached, triggers a log message. The level of logging messages can be adjusted to allow the administrator to determine what kinds of messages are sent to the syslog server. Routers support different levels of logging. The eight levels range from 0 (emergencies), indicating that the system is unstable, to 7 (debugging), which sends messages that include router information.

**Note**: The default level for syslog is 6, informational logging. The default for console and monitor logging is 7, debugging.

a. Use the **logging trap** command to determine the options for the command and the various trap levels available.

```
R1(config)# logging trap ?
<0-7>          Logging severity level
alerts         Immediate action needed           (severity=1)
critical       Critical conditions               (severity=2)
debugging      Debugging messages                (severity=7)
emergencies    System is unusable                (severity=0)
errors         Error conditions                  (severity=3)
informational  Informational messages            (severity=6)
notifications  Normal but significant conditions (severity=5)
warnings       Warning conditions                (severity=4)
<cr>
```

b. Define the level of severity for messages sent to the syslog server. To configure the severity levels, use either the keyword or the severity level number (0–7).

| Severity Level | Keyword | Meaning |
|---|---|---|
| 0 | emergencies | System is unusable |
| 1 | alerts | Immediate action required |
| 2 | critical | Critical conditions |
| 3 | errors | Error conditions |
| 4 | warnings | Warning conditions |
| 5 | notifications | Normal but significant condition |
| 6 | informational | Informational messages |
| 7 | debugging | Debugging messages |

**Note**: The severity level includes the level specified and anything with a lower severity number. For example, if you set the level to 4, or use the keyword **warnings**, you capture messages with severity level 4, 3, 2, 1, and 0.

c. Use the **logging trap** command to set the severity level for R1.

```
R1(config)# logging trap warnings
```

What is the problem with setting the level of severity too high or too low?

If the command **logging trap critical** were issued, which severity levels of messages would be logged?

## Step 4: Display the current status of logging for R1.

a. Use the **show logging** command to see the type and level of logging enabled.

```
R1# show logging
Syslog logging: enabled (0 messages dropped, 3 messages rate-limited, 0 flushes, 0
overruns, xml disabled, filtering disabled)

No Active Message Discriminator.



No Inactive Message Discriminator.


    Console logging: level debugging, 72 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 72 messages logged, xml disabled,
                    filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    Persistent logging: disabled

No active filter modules.

    Trap logging: level warnings, 54 message lines logged
        Logging to 192.168.1.3  (udp port 514, audit disabled,
              link up),
              3 message lines logged,
              0 message lines rate-limited,
              0 message lines dropped-by-MD,
              xml disabled, sequence number disabled
              filtering disabled
        Logging Source-Interface:       VRF Name:
<output omitted>
```

At what level is console logging enabled?

At what level is trap logging enabled?

What is the IP address of the syslog server?

What port is syslog using?

### Step 5: Make changes to the router and monitor syslog results on the PC.

a. Verify that the syslog server is already started on PC-A. Start the server as necessary.

b. To verify that syslog server is logging the message, disable and enable R1's G0/0/0 interface.

```
R1(config)# interface g0/0/0
R1(config-if)# shut
.Jan 31 12:02:50.376: %LINK-5-CHANGED: Interface GigabitEthernet0/0/0, changed state
to administratively down
.Jan 31 12:02:51.376: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0/0, changed state to down
R1(config-if)# no shut
.Jan 31 12:03:11.302: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 192.168.1.4 port
514 started - CLI initiated
.Jan 31 12:03:14.365: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/0, changed state to
up
.Jan 31 12:03:15.365: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0/0, changed state to up
.Jan 31 12:03:59.894: %OSPF-5-ADJCHG: Process 1, Nbr 10.2.2.2 on GigabitEthernet0/0/0
from LOADING to FULL, Loading Done
```

c. Navigate to PC-A to view the syslog messages.

## Router Interface Summary Table

| Router Model | Ethernet Interface #1 | Ethernet Interface #2 | Serial Interface #1 | Serial Interface #2 |
|---|---|---|---|---|
| 1900 | Gigabit Ethernet 0/0 (G0/0) | Gigabit Ethernet 0/1 (G0/1) | Serial 0/0/0 (S0/0/0) | Serial 0/0/1 (S0/0/1) |
| 2900 | Gigabit Ethernet 0/0 (G0/0) | Gigabit Ethernet 0/1 (G0/1) | Serial 0/0/0 (S0/0/0) | Serial 0/0/1 (S0/0/1) |
| 4221 | Gigabit Ethernet 0/0/0 (G0/0/0) | Gigabit Ethernet 0/0/1 (G0/0/1) | Serial 0/1/0 (S0/1/0) | Serial 0/1/1 (S0/1/1) |

| Router Model | Ethernet Interface #1 | Ethernet Interface #2 | Serial Interface #1 | Serial Interface #2 |
|---|---|---|---|---|
| 4300 | Gigabit Ethernet 0/0/0 (G0/0/0) | Gigabit Ethernet 0/0/1 (G0/0/1) | Serial 0/1/0 (S0/1/0) | Serial 0/1/1 (S0/1/1) |

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