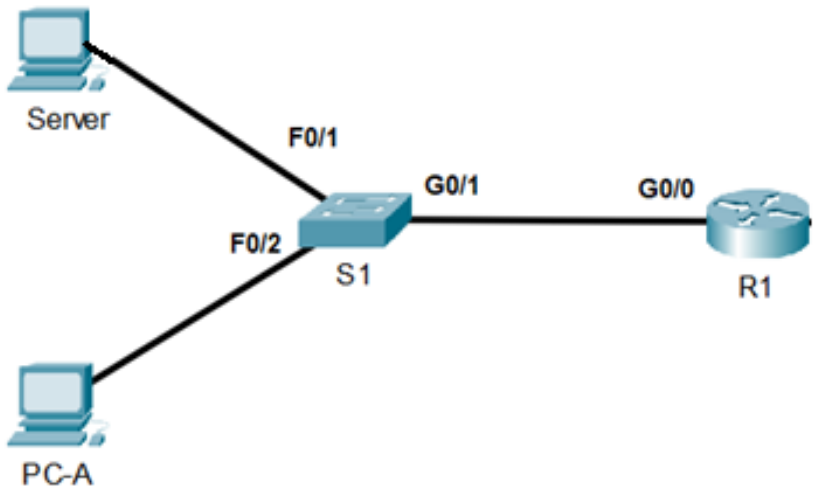
Dave Aldwin Bolima
Aldrich Matthew Go
Bernard Dimero
Shaun Tristan Lim

Lab 6.2 Using Network Management Tools (NTP and Syslog) – F2F

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0	192.168.10.1	255.255.255.0	N/A
S1	VLAN 1	192.168.10.2	255.255.255.0	192.168.10.1
PC-A	NIC	192.168.10.4	255.255.255.0	192.168.10.1
Server	NIC	192.168.10.3	255.255.255.0	192.168.10.1

Objectives

Part 1: Build the Network and Configure Basic Device Setting

Part 2: Configure an NTP Server and Client

Part 3: Configure Syslog

Background / Scenario

Network management is an essential task of a network administrator once a network is deployed and operational. It involves regularly monitoring device operations and making the necessary configuration changes in response to various network scenarios. To reduce the administrative overhead of performing network management, several tools are available. These include device discovery tools, logging, and centralized management. In this lab, you will configure and explore the capabilities of widely used network management tools and protocols: NTP, and Syslog.

Required Resources

- 1 Router
- 1 Switch
- 2 PCs with terminal emulation program and Syslog software, such as Tftpd32
- Console cables to configure the Cisco IOS devices via the console ports
- Ethernet and serial cables as shown in the topology

Instructions

Part 1: Build the Network and Configure Basic Device Settings

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

This part may be skipped if this activity is performed immediately after 6.1 in the same session and all devices in the topology were connected and configured in the prior activity.

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

Step 2: Initialize and reload the network devices as necessary.

Step 3: Configure basic device settings for the switch.

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

b. Enter configuration mode.

c. Configure the hostname and IP address according to the topology.

d. Verify that the switchports with connected Ethernet cables are enabled.

Step 4: Configure basic device settings for the router.

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

b. Enter configuration mode.

c. Configure the hostname and interface IP addresses according to the topology.

Step 5: Configure basic device settings for the PCs

Configure the interface IP addresses according to the topology.

Step 6: Test connectivity

Ensure that devices including PCs, router and switch can ping each other. Troubleshoot as necessary if any does not work.

Part 2: Configure NTP

In Part 2, you will configure R1 as the NTP server and S1 as the NTP client of R1. Synchronized time is important for event logging and debug functions. If the time is not synchronized, it is difficult to determine what network event caused the message.

Step 1: Display the current time.

Issue the **show clock** command to display the current time on R1.

```
R1# show clock
*12:30:06.147 UTC Tue May 14 2013
```

Record the information regarding the current time displayed in the following table.

Date	Jan 1, 1970
Time	00:38:36.767
Time Zone	UTC

Step 2: Set the time.

Use the **clock set** command to set the time on R1. The following is an example of setting the date and time.

```
R1# clock set 9:39:00 05 july 2021
R1#
*Jul  5 09:39:00.000: %SYS-6-CLOCKUPDATE: System clock has been updated from 12:30:54
UTC Tue May 14, 2013 to 09:39:00 UTC Fri Jul 5, 2021, configured from console by
console.
```

Step 3: Configure the NTP master.

Configure R1 as the NTP master by using the **ntp master** *stratum-number* command in global configuration mode. The stratum number indicates the number of NTP hops away from an authoritative time source. In this lab, the number 5 is the stratum level of this NTP server.

```
R1(config)# ntp master 5
```

Step 4: Configure the NTP client.

a. Issue **show clock** command on S1. Record the current time displayed on S1 in the following table.

Date	July 5 2021
Time	9:39:26.119
Time Zone	UTC

b. Configure S1 as the NTP client. Use the **ntp server** command to point to the IP address or hostname of the NTP server. The **ntp update-calendar** command periodically updates the calendar with NTP time.

```
S1(config)# ntp server 192.168.10.1
S1(config)# ntp update-calendar
```

Step 5: Verify NTP configuration.

c. Use the **show ntp associations** command to verify that S1 has an NTP association with R1.

```
S1# show ntp associations
```

```
     address         ref clock       st   when   poll reach   delay   offset    disp
*~192.168.10.1    127.127.1.1      5     11     64   177 11.312   -0.018   4.298
 * sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
```

d. Issue **show clock** on R1 and S1 to compare the timestamp.

Note: It could take a few minutes before the timestamp on S1 is synchronized with R1.

```
R1# show clock
09:43:32.799 UTC Mon April 5 2021
S1# show clock
09:43:37.122 UTC Mon April 5 2021
```

Part 3: Configure Syslog

Syslog messages from network devices can be collected and archived on a syslog server. In Part 3, you will configure S1 and R1 to send log device events remotely to the syslog service running on the Server host.

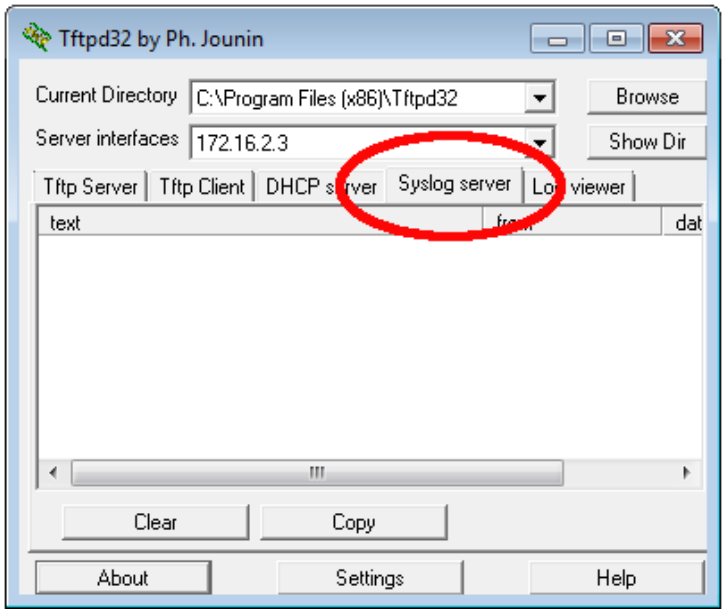
Step 1: Install a syslog service (Optional)

If a syslog service is not already installed on the Server PC, download and install the latest version of a syslog server, such as Tftpd32, on the PC. The latest version of Tftpd32 can be found at the following link:

http://tftpd32.jounin.net/

Step 2: Start the syslog service on Server.

After starting the Tftpd32 application, click the **syslog server** tab.



Step 3: Enable the timestamp service on R1.

Turn on the timestamp service for logging on R1.

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

Step 4: Verify logging configuration on R1

Issue the **show logging** command to display the logging status on R1.

```
R1# show logging
Syslog logging: enabled (0 messages dropped, 2 messages rate-limited, 0 flushes, 0
overruns, xml disabled, filtering disabled)

No Active Message Discriminator.


No Inactive Message Discriminator.


    Console logging: level debugging, 47 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 47 messages logged, xml disabled,
                    filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    Persistent logging: disabled

No active filter modules.
```

Based on the output, what logging destinations are currently enabled on R1?

Console logging is enabled and is set to the debugging level. Similarly, monitor logging is enabled and is also set to the debugging level. However, buffer logging is disabled.

What is the message logging trap level set for these log destinations?

The message logging trap level is set to the informational level for the enabled log destinations on R1. This means that the system will log events that are the result of normal operations, in addition to error messages, critical, alert, emergency, warning, and notice messages.

Step 5: Configure R1 to log messages to the syslog server.

a. Ensure that R1 can reach the Server PC. Do a test ping to verify.

b. Configure R1 to send Syslog messages to the Server. The IP address of the server is 192.168.10.3.

```
R1(config)# logging host 192.168.10.3
```

c. Verify that remote logging for R1 is now enabled

```
R1# show logging
R1# show logging
Syslog logging: enabled (0 messages dropped, 2 messages rate-limited, 0 flushes, 0
overruns, xml disabled, filtering disabled)

No Active Message Discriminator.
```

```
No Inactive Message Discriminator.


    Console logging: level debugging, 47 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 47 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 informational, 49 message lines logged
        Logging to 172.16.2.3  (udp port 514, audit disabled,
              link up),
              6 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:
```

What is the IP address of the server as indicated in the output?	192.168.10.3
What transport protocol and port number is used for logging?	UDP
What message logging trap level set for remote logging?	Informational

Step 6: Configure and observe the effect of logging severity levels on R1.

e. Use the **logging trap ?** command to determine the various trap levels availability. When configuring a level, the messages sent to the syslog server are the trap level configured and any lower levels.

```
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>
```

If the **logging trap warnings** command was issued, which severity levels of messages are logged?

The level warning are from 0 to 4.

f. Change the logging severity level to 4.

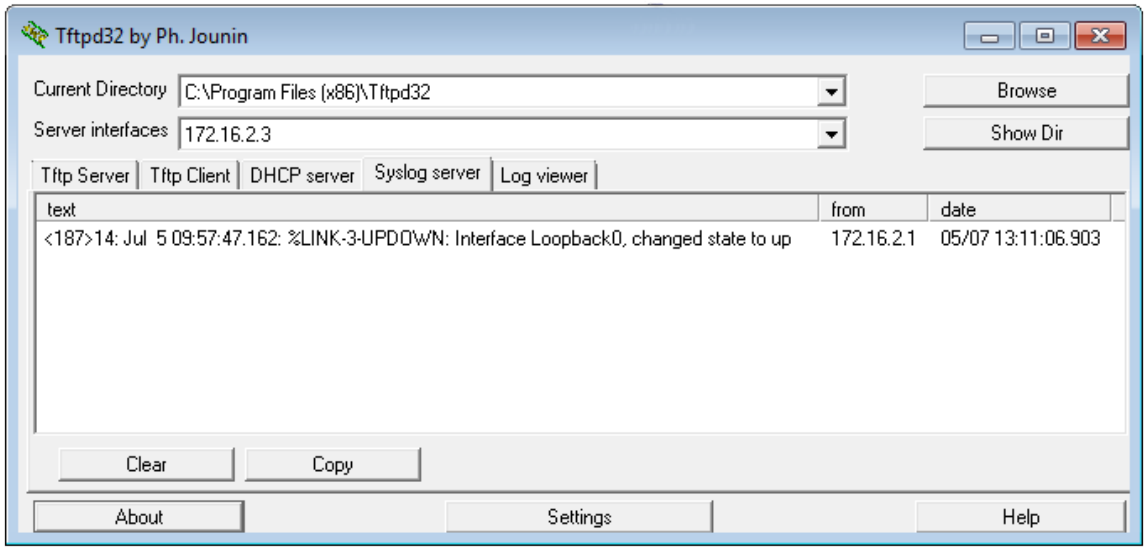
```
R1(config)# logging trap warnings
```

or

```
R1(config)# logging trap 4
```

g. Create interface Loopback0 on R1 and observe the log messages on both the terminal window and the syslog server window on the Server PC.

```
R1(config)# interface lo 0
R1(config-if)#
Jul  5 09:57:47.162: %LINK-3-UPDOWN: Interface Loopback0, changed state to up
Jul  5 09:57:48.162: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0,
changed state to up
```



h. Remove the Loopback 0 interface on R2 and observe the log messages.

```
R1(config-if)# no interface lo 0
R1(config)#
Jul  5 10:02:58.910: %LINK-5-CHANGED: Interface Loopback0, changed state to
administratively down
Jul  5 10:02:59.910: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0,
changed state to down
```

At severity level 4, are there any log messages on the syslog server? If any log messages appeared, explain what appeared and why.

Yes. What appeared was <187> 30: Jul 5 10:09:19:907: %LINK-3-UPDOW… 192.168.10.1 01/03 15:46:28. This appeared because we added the loopback interface. However, it only displays up since the logging trap was set to 4, so shutting down did not appear in the syslog.

i. Change the logging severity level to 6.

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

or

```
R1(config)# logging trap 6
```

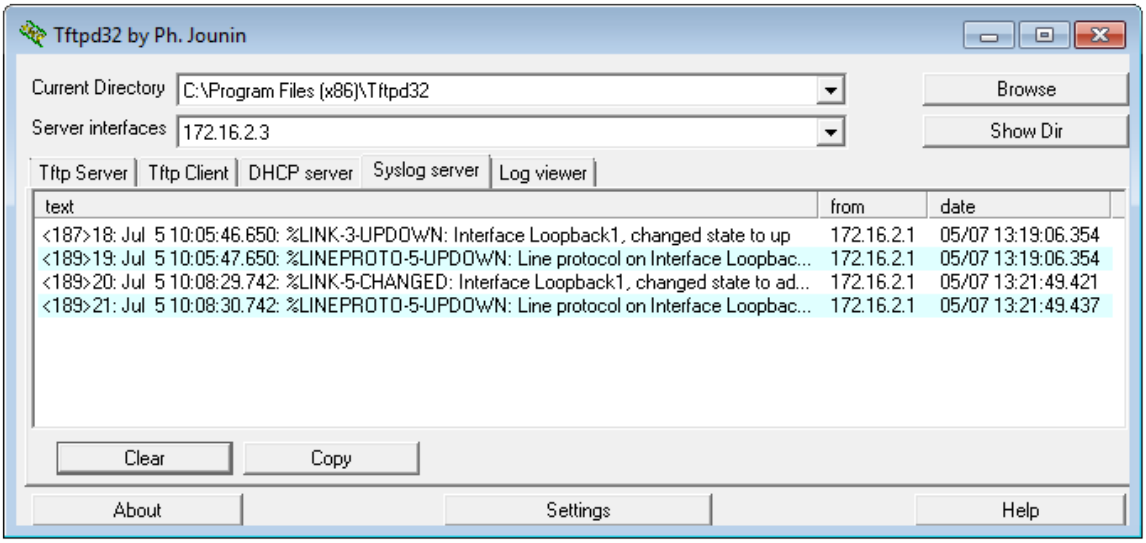
j. Clear the syslog entries on the Server PC. Click **Clear** in the Tftpd32 dialog box.

k. Create the Loopback 1 interface on R1.

```
R1(config)# interface lo 1
Jul  5 10:05:46.650: %LINK-3-UPDOWN: Interface Loopback1, changed state to up
Jul  5 10:05:47.650: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1,
changed state to up
```

l. Remove the Loopback 1 interface from R1.

```
R1(config-if)# no interface lo 1
R1(config-if)#
Jul  5 10:08:29.742: %LINK-5-CHANGED: Interface Loopback1, changed state to
administratively down
Jul  5 10:08:30.742: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1,
changed state to down
```



m. Observe the syslog server output. Compare this result with the results at trapping level 4. What is your observation?

In logging trap level 6, both bringing up the interface and shutting down the interface was recorded. Unlike in logging trap l;evel 4, where it only shows bringing the interface up.

Step 7: Configure S1 to log remotely to the server

Perform the necessary configurations on S1 so that it also uses timestamps and logs it events to the Server.

What commands are needed to accomplish this?

S1(config)# service timestamps log datetime msec **S1(config)# logging trap informational**

Reflection Questions

1. What are the advantages of using NTP and remote logging through Syslog together to perform device monitoring tasks for network management?

   The combination of Network Time Protocol (NTP) and Syslog for device monitoring tasks in network management offers several advantages. NTP ensures that all devices in a network have synchronized clocks, which is crucial for accurately timestamping the log messages generated by these devices. These timestamped Syslog messages allow events from different sources to be organized chronologically, providing a clear view of network communication processes. In terms of security, NTP can reduce the susceptibility of your systems to virus attacks and intrusion from hackers. Syslog can also help detect and respond to security incidents by logging relevant events.

2. What issues can be caused by setting the level of severity too high (lowest level number) or too low (highest level number) for syslog?

   It may lead to missing crucial details, this can be misinterpreted when setting it to low (Level 0 or Level 1)

   While if we were to set it to too high (level 6 or level 7), it can flood the logs with excessive information, consuming resources and impacting the performance. We should manually adjust this for specific needs, security considerations, and for the compliance requirements that the company requests.