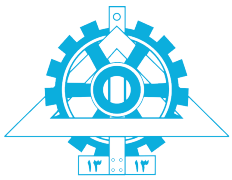
Bridges, LANs and the Cisco IOS[1]

HARDNESS : 6/10



University of Tehran
School of Electrical and Computer Engineering

دانشگاه تهران
دانشکده‌ی مهندسی برق و کامپیوتر

Computer Network Lab

آزمایشگاه شبکه‌های کامپیوتری

Professor:
Dr. Ahmad Khonsari
دکتر احمد خونساری
a_khonsari@ut.ac.ir

Amir Haji Ali Khamseh'i
امیر حاجی‌علی‌خمسه‌ء
khamse@ut.ac.ir

Reza Sharifnia
رضا شریف نیا
Reza.sharifnia@ut.ac.ir

Muhammad Borhani
محمد برهانی
borhani.m@ut.ac.ir

AmirAhmad Khordadi
امیراحمد خردادی
a.a.khordadi@ut.ac.ir

Sina Kashipazha
سینا کاشی‌پزها
sina_kashipazha@ut.ac.ir

Hadi Safari
هادی صفری
hadi.safari@ut.ac.ir

November 9, 2021

۱۸ آبان ۱۴۰۰

[1]S. Panwar, S. Mao, J.-dong Ryoo, and Y. Li, "Bridges, LANs and the Cisco IOS," in TCP/IP Essentials: A Lab-Based Approach, Cambridge: Cambridge University Press, 2004, pp. 61–76.

Objectives

- The Cisco Internet Operating System (IOS) software.
- Configuring a Cisco router.
- Transparent bridge configuration and operation.
- The spanning tree algorithm.

Part I

Exercises on Cisco IOS

In this lab, you need two hosts, a bridge, and two hubs, which are required to be connected as shown in Figure 3.7, Table 3.2 and Table 3.3.

1 Network Setup

In this exercise we build the connection to the router (see Figure 3.7, Table 3.2 and Table 3.3).
Use two hosts, two hubs, and one router (as a bridge). Then, connect first host to the first hub and connect the first port of the bridge to the previously mentioned hub. Now, connect the other side of the topology just like first side.

Table 1: Router and Host IP addresses for Figure 3.7 (Table 3.2, Table 3.3)

Router		$Host_A$		$Host_B$	
eth0	eth1	Name	IP Address	Name	IP Address
128.238.61.1/24	128.238.61.2/24	h0	128.238.61.100/24	h1	128.238.61.101/24

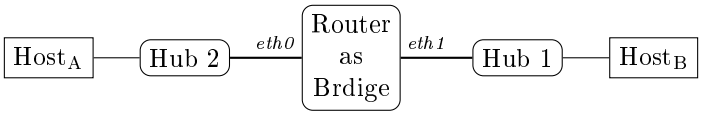


Figure 1: Using a transparent bridge (Figure 3.7)

2 IOS Console

Open Router Console (In **GNS3** right click on router and open console window). You should now be in the *Privileged EXEC* mode.

Type `help` to learn how to use the online help. Study Figure 3.6 of reference book. Navigate through the *User EXEC*, *Privileged EXEC*, *Global Configuration*, and *Interface Configuration* modes. In each mode, type `?` to display a list of available commands and study these commands (below commands).

help command output:

$R_1\#$

```
help
```

? command in User EXEC mode output:

$R_1\#$

```
disable
```

$R_1>$

```
?
```

? command in Privileged EXEC mode output:

$R_1>$

```
enable
```

$R_1\#$

```
?
```

? command in Global Configuration mode output:

$R_1\#$

```
conf term
```

$R_1(\text{config})\#$

```
?
```

? command in Interface Configuration mode output:

$R_1(\text{conf})\#$

```
int f0/0
```

$R_1(\text{config-if})\#$

```
?
```

Type `show version` in the *User EXEC* mode to display the Cisco IOS banner (below commands). Identify which Cisco IOS Release is running in the router. Save the Cisco IOS banner for your lab report.

$R_1(\text{conf-if})\#$

```
Ctrl+Z
```

$R_1\#$

```
disable
```

$R_1>$

```
show version
```

See the Cisco IOS banner. Identify the release of the Cisco IOS software in the router.

Part II

A Simple Bridge Experiment

Figure 3.7 shows a simple case of the use of bridges, which consists of two network segments connected by a bridge. With this simple topology, we can easily capture initial BPDUs before each bridge is engaged in the spanning tree calculation.

Configure transparent bridging as in Figure 3.7, Table 3.2 and Table 3.3. Note that the default configuration of the hosts and the bridges are different from those in the tables. You need to change the IP addresses of the bridge interfaces,[1] as well as set the bridge group and enable the spanning tree algorithm (see the previous section on bridge configuration). Do the following experiments.

Config Cisco Router as transparent Bridge:

$R_1\#$

```
config term
    no ip routing
    bridge 1 protocol ieee
    int f0/0
        ip addr 128.238.61.1 255.255.255.0
        bridge-group 1
        no shut
        exit
    int f0/1
        ip addr 128.238.61.2 255.255.255.0
        bridge-group 1
        no shut
        end
Ctrl+Z
```

3 Bridge Packet

Run below commands on *h0* and *h1* machine:

h_0's Console

```
tcpdump -en ip proto 1
```

h_1's Console

```
tcpdump -en ip proto 1
```

Send `ping` messages to *h1* machine:

h_0's Auxiliary console

```
ping -sv 128.238.61.101
```

After receiving the tenth echo reply, quit the `ping` process, and save the `tcpdump` outputs from both machines.

During this exercise, don't run `ping` programs at the same time from other host. For clean results, do your experiments in turn.

[1] As soon as you change the IP address of the bridge interface your host is connected to, the `telnet` connection will be lost. You need to again change the IP address of your machine to be in the same subnet as the bridge interface. See Section 3.3.3 of reference book.

Report

1. What are the IP and MAC addresses of a packet that went from your machine to the bridge? What are the IP and MAC addresses of a packet that went from the router to your partner's machine?
2. Answer the same questions, but for the echo reply that was returned from your partner's machine.
3. Using the `tcpdump` outputs from both machines, calculate the average delay that a packet experienced in the bridge. [2]

   Note: The calculated time is also used in the next class session. Please write it down.

Show all the steps and submit the `tcpdump` outputs with your report.

4 STP/BPDU Packet

Run below commands on all machines to capture 2 BPDUs messages generated by the bridge. Save the BPDUs for the lab report.

h_0's Console

```
tcpdump -e -c 5 ether multicast -vv
```

h_1's Console

```
tcpdump -e -c 5 ether multicast -vv
```

explain stp and bpdu

You should collect BPDUs in this exercise. These BPDUs will be helpful when studying the spanning tree algorithm later in this chapter.

Report

1. How frequently (in seconds) does a bridge sends its BPDUs?
2. Submit the two different BPDUs you saved. Identify the values of root ID, root path cost, bridge ID, and port ID for each BPDU[3] (may need to check BPDU message format Figure 3.4).

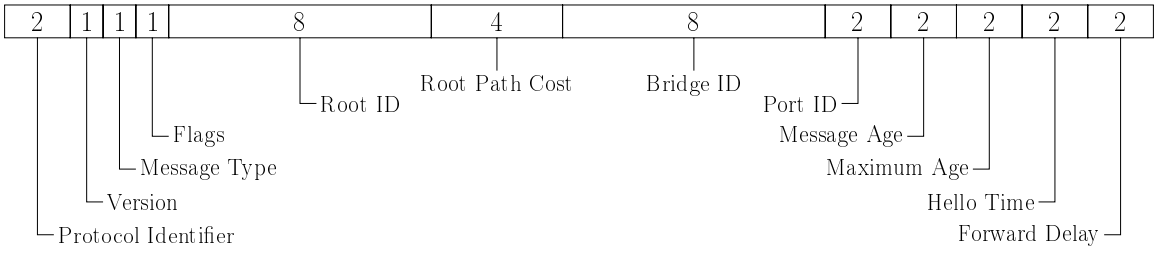


Figure 2: BPDU message format (Figure 3.4)
The numbers indicate the field length in byte.

[2] Note that the system times of the two machines might be different.
[3] You may check the physical addresses of network interfaces. You need the MAC addresses to help analyze the BPDUs.

Part III

Spanning Tree Exercises

In this section, we will use Figure 3.8 as our network topology. You need to change the IP addresses of the bridge interfaces, as well as that of your machines. Refer to Section 3.3.4 of reference book on how to configure a transparent bridge. Also see Section 3.3.3 of reference book on how to handle a frozen telnet session after you change the bridge IP address.

Upon being started, a transparent bridge learns the network topology by analyzing source addresses of incoming frames from all attached networks. The next exercise shows the process by which a transparent bridge builds its filtering database.

You can read Appendix A and Appendix B to learn how Spanning Tree Algorithm work.

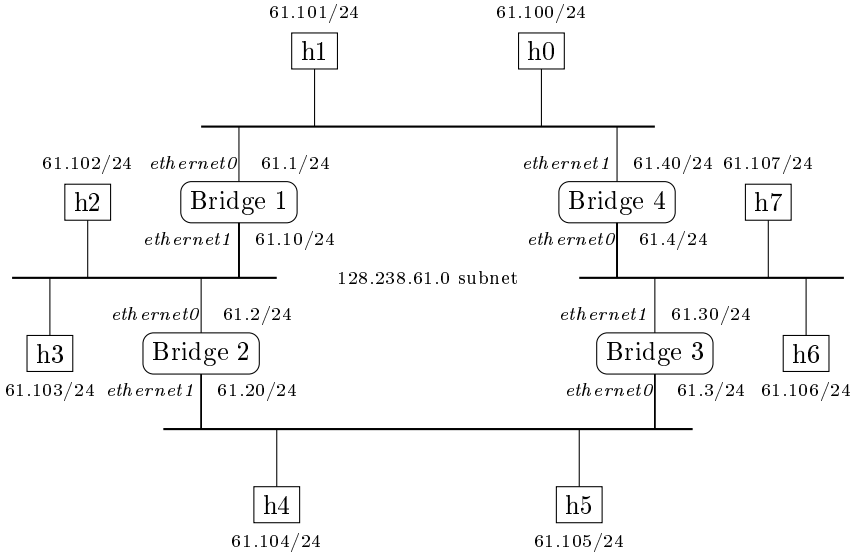


Figure 3: Bridge experiment network (Figure 3.8)

5 Multi Bridge Path

After configuring the network in Figure 3.8 (Note: in Figure 3.8 that downloaded from Github, the network is configured), login to the bridge 1 (Note: right click on R1 and select Console item from Pop-up menu).

Get to the *Privileged EXEC*[4] mode (is activated by default in **GNS3**). Type `show bridge` to see the entries in the bridge forwarding database.

$R_1\#$

```
show bridge
```

Whenever you `ping` or `telnet` from *h0* to a host that is not in the table (table form bridge 1), observe how the filtering database in the bridge is expanded.

h_0's Console

```
ping 128.238.61.102
Press Ctrl+C to stop this command
```

h_0's Console

[4] use **enable** command to active *Privileged EXEC*

```
ping 128.238.61.104
Press Ctrl+C to stop this command
```

h_0's Console

```
ping 128.238.61.106
Press Ctrl+C to stop this command
```

R_1,R_2 ,R_3 and R_4#

```
show bridge
```

Note: You may use the `clear bridge group` command to remove any learned entries from the filtering database, if you see a full filtering database or **if you want to repeat the above exercise**.

R_1#

```
clear bridge 1
```

Report

1. From the output of `show bridge` , identify which bridge ports are blocked, and which ports are in the forwarding state for each bridge.

6 STP Process

Using below command on all LAN segments, capture the BPDU packet flowing on your network segment.

h_0, h_2, h_4 and h_6's Console

```
tcpdump -ex ether multicast -vv
```

Note: To collect all BPDU packets from start time, you need restart (reload) all router. for restart (reload) all router, you can right click on routers and click on stop and then start.

Login to each bridge (open GNS3 router console) to collect the `show bridge` outputs.

R_1#

```
show bridge
```

Report

1. Submit the four different BPDUs (from four network sections) you saved. Identify the values of root ID, root path cost, bridge ID, and port ID for each BPDU.
2. Based upon the initial BPDUs saved in Spanning Tree Exercises, draw the spanning tree seen by the BPDUs (or explain root node and its child). Identify the root ports and the root path cost (in hop counts) for each bridge. Based on Spanning tree algorithm section in the reference book (section 3.2.3 page 63), explain how root node selected?
3. Write the final BPDUs you collected using the three-tuple format: *root ID, root path cost, bridge ID.*
4. Once you have the spanning tree, justify it using the four final BPDUs collected in this exercise and/or the output of the `show bridge` command.

7 STP over Topology Dynamics

First, send `ping` messages from *h3* to *h4*, while `tcpdump` is running. Let the two programs run during this exercise.

h_3's Console

```
tcpdump
```

h_3's Auxiliary console

```
ping 128.238.61.104
```

Then, disconnect the cable from the *ethernet0* port of *Bridge2* (in GNS3, right click on link and select suspend) from the hub (or shut the router interface), and get system time (`date` , `date '+%D %T.%N'` or `date +%s%N` to get nano seconds) on *h3* or *h4* to get the current time.

Observe the `ping` and `tcpdump` windows. When the connection is reestablished, get the `time` again. How long does it take the spanning tree algorithm to react to the change in the topology?

Note: The calculated time is also used in the next class session. Please write it down.

Once you can successfully reach other hosts, get to the bridges to run `show bridge` to collect the port states. Also collect BPDUs from all the LAN segments as you did in the previous exercise.

R_1, R_2, R_3 and R_4#

```
show bridge
```

After every student has collected the required data, connect the cable to the original position (in GNS3, right click on link and select Resume). Again, measure the time it takes for the bridges to adapt to the new change.

Report

1. Draw the new tree formed after the cable was disconnected, based on the BPDUs you collected in this exercise. Specify the state of each bridge port.

Part IV

Exercise on the Cisco IOS Web Browser UI

8 Cisco IOS HTTP REST API

In this section we create simple network with a router and two gui host with *Internet Browser*. (Fig-3_gui in Github Figures folder)

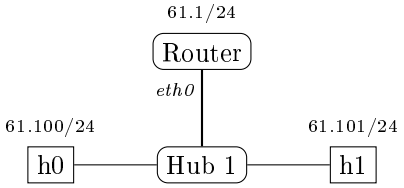


Figure 4: Two gui host with router (Fig-3_gui)

First, you must configure a router using the web browser UI. To enable the web server, login to the router (open GNS3 router console) and config as below in the *Global Configuration* mode.

$R_1\#$

```
config term
    R1(config)#aaa new-model
    R1(config)#aaa authentication login default local
    R1(config)#aaa authorization exec default local
    R1(config)#username netlab privilege 15 password netlab
    R1(config)#ip http server
    R1(config)#ip http secure-server
    R1(config)# ip http authentication local
```

Next, start a web browser (e.g. *Mozilla* in Linux) in *h0* or *h1*. So, double-clicking on the *host* node or right-click on the *host* and select `Console`. This will open a VNC connection to the *client* on your machine's web browser. (Depending on your computer configuration, this may take some times).

Note: If a page titled *Warning: Potential Security Risk Ahead* is displyed for the VNC connection, Click on *Advanced* and then click *Accept the Risk and Continue* button (Figure 5).

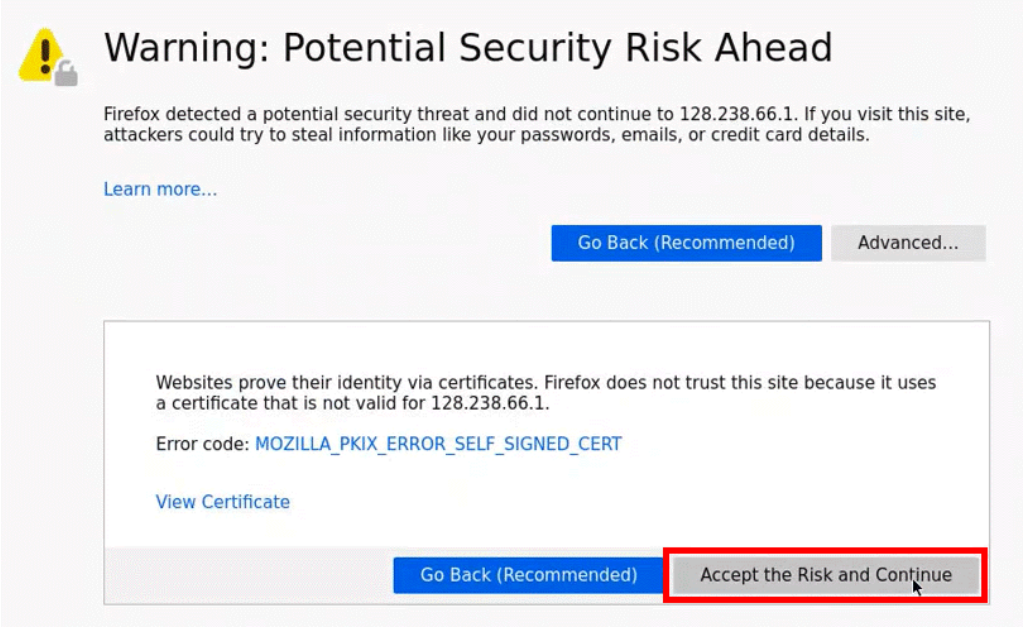


Figure 5: Warning page due to unknown certificate

On the client, start *Mozilla Firefox* web browser (You can open *Mozilla Firefox* web browser from the client's start menu (Figure 6).

Then, enter the IP address of the router interface in the client's browser address bar:

client's `Firefox`

```
http://128.238.61.1
```

When prompted, enter netlab for user name and password. Then you can browse the router configuration web pages and configure the router there.

CAUTION: Pay attention to distinguish the *client's Firefox* from your machine's web browser (Figure 7). All URL navigations must occur at the *client's Firefox*.

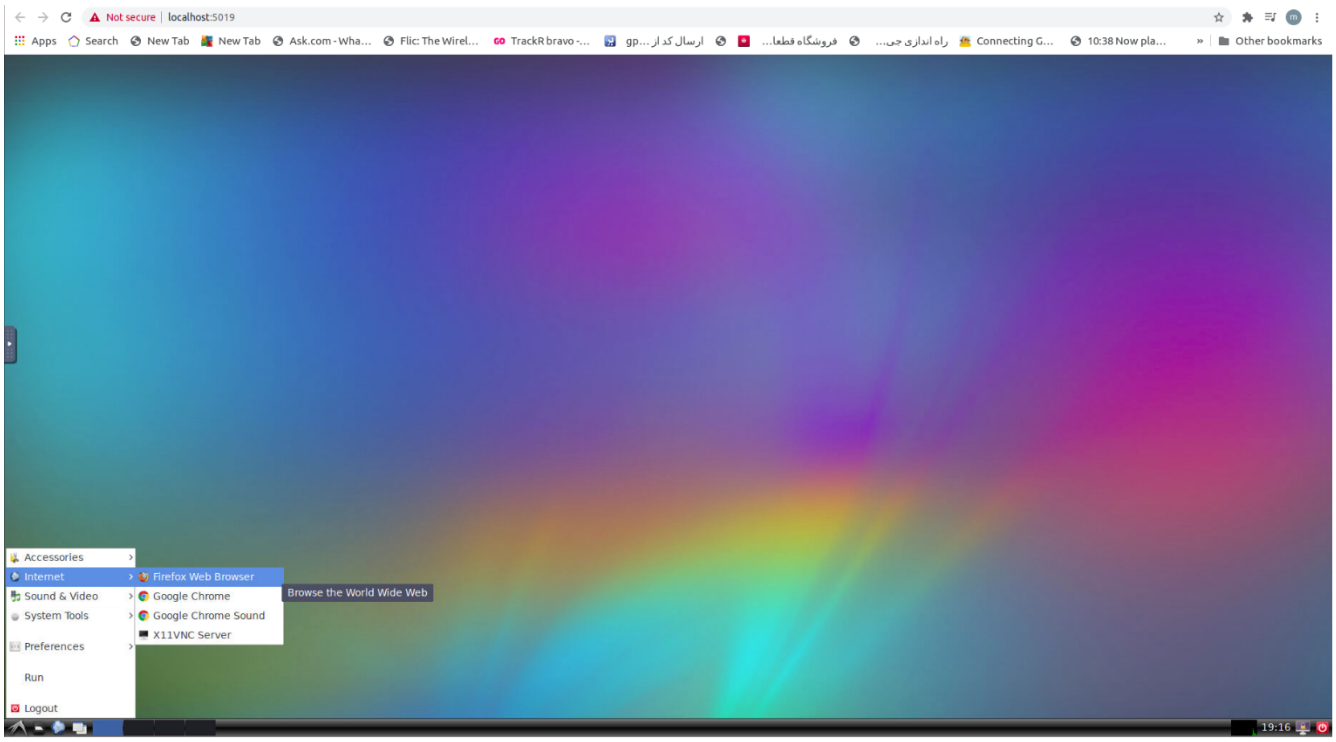


Figure 6: Start Menu → Internet → Firefox Web Browser

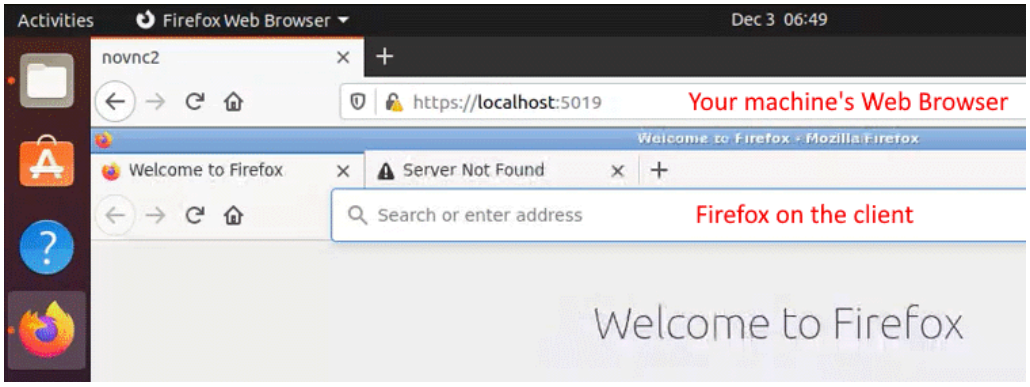


Figure 7: *client's Firefox* inside your machine's Web Browser

A Spanning Tree Algorithm

The spanning tree algorithm defined in the IEEE 802.1d standard is used in bridged networks to build trees dynamically. It works as follows.

1. Each bridge is assigned a unique identifier, and each port of a bridge is assigned an identifier unique to that bridge. Typically, the identifier of a bridge is a priority concatenated with one of the bridge ports' MAC address, and the identifier of a port is a priority concatenated with a port index local to the bridge. Each bridge port has a corresponding path cost, which indicates the cost to transfer a frame to an attached network segment through that port.

2. Select the root bridge, which is the one with the lowest-value bridge identifier. The ID of the root is called the root ID.

3. Each bridge selects its root port. The root port of a bridge is the port from which the root bridge can be reached with the least aggregate path cost (called the root path cost).

4. Determine the designated bridges and the designated ports. Each network segment is associated with a designated bridge, which provides the shortest path to the root bridge and is the only bridge allowed to

forward frames to and from the root. The port connecting a designated bridge to the network segment is a designated port. If more than one bridge provides the same root path cost, the bridge with the lowest-valued bridge identifier is selected as the designated bridge.

5. Only the root ports and designated ports of the bridges are allowed to forward frames. All other bridge ports are blocked.

6. The above steps are repeated whenever the network topology changes.

B Bridge Protocol Data Units

To implement the spanning tree algorithm in a distributed manner, bridges exchange configuration information using a message called bridge protocol data units (BPDUs). The format of a BPDU message is given in Fig 3.4 with the definition of the fields given below.

- *Protocol Identifier*, *Version*, and *Message Type*: These three fields are always set to 0.
- *Flags*: The least significant bit, called the Topology Change (TC) bit, is set to signal a topology change. The most significant bit is to acknowledge receipt of a BPDU with the TC bit set. The remaining six bits are not used.
- *Root ID*: Identifies the root bridge by listing its 2-byte priority followed by a 6-byte Ethernet address. The priority value can be set in the Global Configuration mode. The default priority is 0x8000.
- *Root Path Cost*: The path cost to the root bridge.
- *Bridge ID*: The identifier of the bridge sending the message. Port ID: Each bridge port has a unique 2-byte identifier. The first byte is the priority, which is configurable, while the second byte is a number assigned to the port.
- *Message Age*:1 Specifies the amount of time since the root originally sent the BPDU on which the current configuration message is based.
- *Maximum Age*:1 Indicates when the spanning tree topology is recalculated if a bridge does not hear BPDUs from the root bridge. The default value is 15 seconds.
- *Hello Time*:1 Provides the time period between two BPDUs from the root bridge. The default value is 1 second.
- *Forward Delay*:1 provides the amount of time that bridges should wait before switching a port from the blocking state to forwarding state. If a bridge port switches state too soon, not all network links may be ready to change their state, and loops can occur. The default value is 30 seconds.