LAB FOUR – LAN SWITCHING

A *bridge* or *LAN switch* is a device that interconnects two or more local area networks (LANs) and forwards packets between these networks. Different from IP router, bridges and LAN switches operate at the data link layer. For example, bridges and LAN switches forward packets based on MAC addresses, whereas IP routers forward packets abased on IP addresses.

LAN switches are widely deployed in enterprise networks, including university campus networks. Many enterprise networks primarily use LAN switches to interconnect LANs, using IP routers only to connect the enterprise network to the public Internet.

The term *bridge* was coined in the early 1980s. Today, when referring to data link layer interconnection devices, the terms *LAN switch* or *Ethernet switch* (in the context of Ethernet) are much more common. Since many of the concepts, configuration commands, and protocols for LAN switches in Lab 6 use the old term *bridge*, we will, with few exceptions, refer to LAN switches as *bridges*.

This lab covers the main concepts of LAN switching in Ethernet networks: how packets are forwarded between LANs and how the routes of packets are determined. In Part 1 you learn how to configure a Cisco router as a bridge. Parts 2, 3, and 4 explore how forwarding tables of bridges are set up. You learn about the concepts of *learning bridges* and *transparent bridges*, as well as the operation of the spanning tree protocol that enables loop-free routing between interconnected LANs. Part 5 of the lab explores issues that arise when IP routers and bridges operate in the same network.

In this lab, we will first show you how to configure a Cisco router to operate as a bridge. All bridges henceforth in this lab will be Routers configured as bridges. In Part 5 we will use both routers and bridges (routers operating at bridges).

**NOTE:**

When you are working on this lab, it is best to plan enough time to do Part 3 and Part 4 consecutively as Part 4 uses information from Part 3.

# PART 1. Configuring a Cisco Router as a Bridge

The network configuration for Part 1 is shown in Figure 4.1 and Table 4.1. Router1 is set up as a bridge.

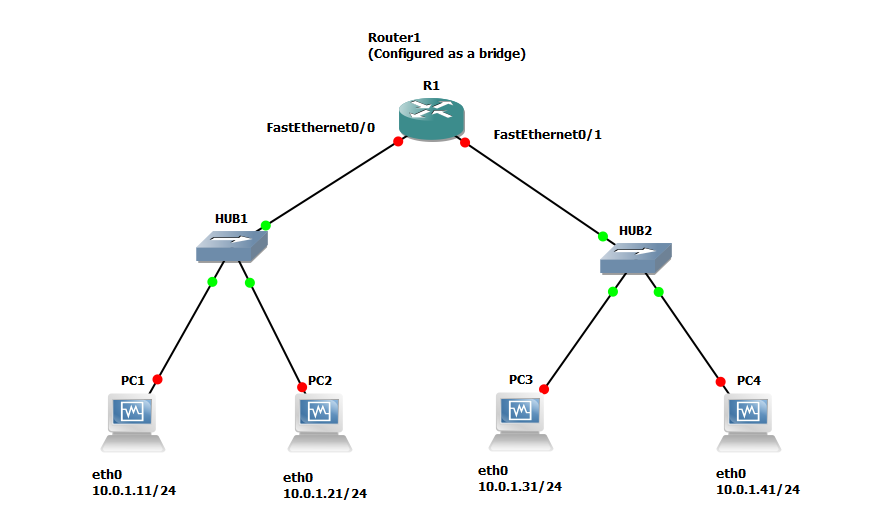


Figure 4.1 Network topology for Part 1.

|  |  |
| --- | --- |
| **PCs** | **eth0** |
| PC1 | 10.0.1.11 / 24 |
| PC2 | 10.0.1.21 / 24 |
| PC3 | 10.0.1.31 / 24 |
| PC4 | 10.0.1.41 / 24 |

Table 4.1 IP addresses of the PCs

### Exercise 1(A). How to Configure a Cisco router to act as a bridge

A Cisco router can be configured to perform the functions of multiple independently operating bridges. This is done by defining a bridge group, which is identified by a number, and associating two or more network interfaces with each bridge group. Packets are forwarded only between interfaces that are assigned to the same bridge group. Since the exercises in Lab 6 use only on bridge group, we always use 1 to identify the group.

|  |
| --- |
| **IOS MODE: GLOBAL CONFIGURATION**  bridge 1 protocol ieee  Defines a bridge group and signs the spanning tree protocol as defined in the IEEE 802.1d Standard to bridge group 1. After the command is issued, the Cisco router forwards packets between all interfaces that are assigned to bridge group 1. A bridge group can be any number between 1 and 63. After defining a bridge group, one can assign network interfaces to the bridge group. It is possible to define multiple bridge groups. In Lab 6, only one bridge group (with identifier 1) is used.  bridge 1 priority 128  Assigns the priority 128 to bridge group 1. The priority of a bridge group plays a role in the spanning tree protocol, which is covered in Part5. |

|  |
| --- |
| **IOS MODE: INTERFACE CONFIGURATION**  bridge-group 1  Assigns this network interface to bridge group 1  no bridge-group 1  Removes this network interface from bridge group 1.  bridge-group 1 spanning-disabled  Disables the spanning tree protocol on this interface for bridge group 1  no bridge-group 1 spanning-disabled  Enables the spanning tree protocol on this interface for bridge group 1 |

Once a Cisco router is configured as a bridge, the commands in the next list can be used to display the status of the bridge.

|  |
| --- |
| **IOS MODE: PRIVILEGED EXEC**  show bridge  Displays the entries of the MAC forwarding table  show spanning-tree brief  Displays the spanning tree topology information known to this bridge  show interfaces  Displays statistics of all interfaces, including the MAC addresses of all interfaces |

The commands in the next list disable bridging functions on a Cisco router.

|  |
| --- |
| **IOS MODE: PRIVILEGED EXEC**  no bridge 1  Deletes the defined bridge group. After the command is issued, the cisco router stops forwarding packets between interfaces that are assigned to bridge group 1.  clear bridge  Removes all entries from the MAC forwarding table  clear arp-cache  Clears the ARP table. |

1. **Configuring R1:** Use the following commands to configure Router1 as a bridge:

**R1**> enable

**R1**# configure terminal

**R1**(config)# no ip routing

**R1**(config)# bridge 1 protocol ieee

**R1**(config)# bridge 1 priority 128

**R1**(config)# interface FastEthernet0/0

**R1**(config-if)# bridge-group 1

**R1**(config-if)# bridge-group 1 spanning-disabled

**R1**(config-if)# no shutdown

**R1**(config-if)# interface FastEthernet1/0

**R1**(config-if)# bridge-group 1

**R1**(config-if)# bridge-group 1 spanning-disabled

**R1**(config-if)# no shutdown

**R1**(config-if)# end

**R1**# clear bridge

**R1**# clear arp-cache

The commands disable IP forwarding and set up R1 as a bridge that runs with priority 128. Both Ethernet interfaces are assigned to the bridge, **AND** the spanning tree protocol is disabled.

1. Once R1 has been configured as a bridge, configure the PCs as shown in Figure 4.1 with the IP addresses given in Table 4.1.
2. Delete all entries in the ARP caches of all PCs.
3. Start 2 Wireshark captures, one on the link between bridge (R1) and Hub1 and the other on link between bridge (R1) to Hub2.
4. Issue a ping command from PC1 to PC2, and from PC1 to PC3:

**PC1%** ping 10.0.1.21 -c 10

**PC1%** ping 10.0.1.31 –c 10

1. Use the show bridge command on R1 to show the bridge forwarding table. Screenshot and save.
2. Use the command arp -a to show the arp tables of PC1, PC2 and PC3. Screenshot and save.
3. Stop Wireshark captures and save the output.
4. Run traceroute from PC1 to PC3. Screenshot the output and save.

**PC1%** traceroute 10.0.1.31

### Lab Questions

* From the saved Wireshark output, comment on the route of the ARP and the ICMP packets for both pings.
* Using the output of the show bridge command on R1, can you tell what the path is for the ICMP packets between the hosts?
* From the traceroute output, what is the recorded path between PC1 to PC3?
* What can you tell from the output of the arp -a command on each PC?
* Can you explain the difference in what the show bridge command shows as the path and what the traceroute and arp -a command show?

# PART 2. Learning Bridges

Each bridge has a MAC forwarding table that determines the outgoing port for a packet. When a packet arrives, the bridge looks up the destination MAC address of the packet in its MAC forwarding table and retrieves the outgoing port for this packet. If the destination MAC address is not found in the MAC forwarding table, the bridge floods the packet on all ports, with the exception of the port where the packet arrived on.

Bridges update their forwarding table using what is called a learning algorithm, which works as follows. A bridge examines the source MAC address of each packet that arrives on a particular port and memorizes that the source address is reachable via that port. This is done by adding the source MAC address and the port to the forwarding table. The next time the bridge receives a packet that has this MAC address as destination, the bridge finds the outgoing port in its forwarding table. Bridges that run this algorithm are referred to as learning bridges. All currently deployed Ethernet switches execute the learning algorithm.

An entry in the bridge forwarding table is deleted if is not used (looked up) for a certain amount of time. The maximum time that a MAC address can stay in the forwarding table without a lookup is determined by the *ageing* value, which is a configuration parameter.

Here you investigate the learning algorithm of bridges. The network configuration is as shown in Figure 4.2. The IP addresses for the PCs are the same as in the previous exercise (Table 4.1.).

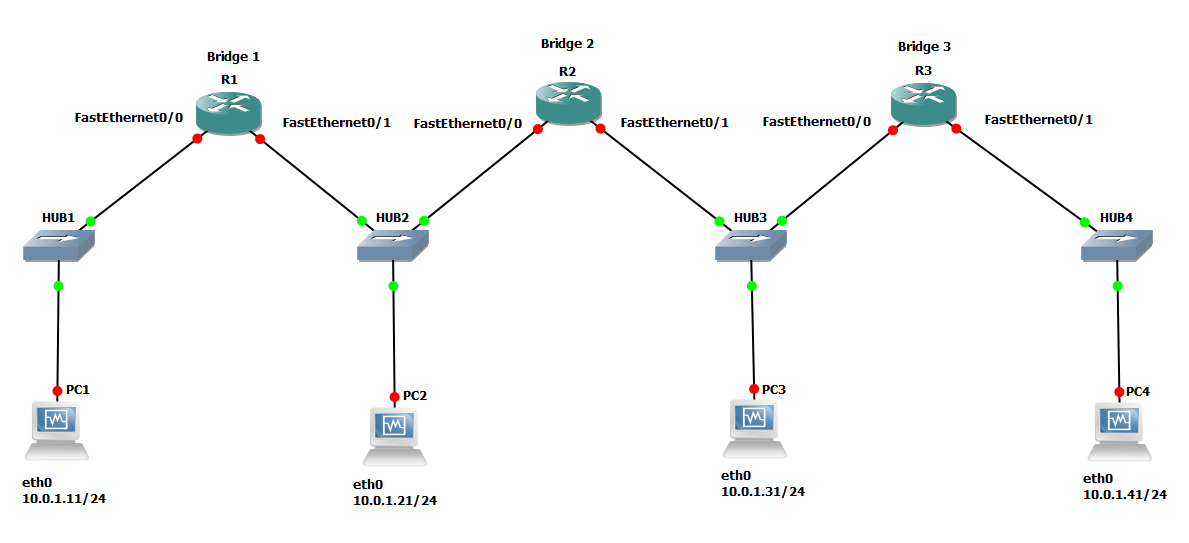


Figure 4.2 Network topology for Part 2.

### Exercise 2(A). Exploring the learning algorithm of bridges

In this exercise you study how bridges set up their MAC forwarding tables from the network traffic.

1. Set up the network configuration as shown in Figure 4.2 and Table 4.1.
2. Configure R1, R2 and R3 as bridges. Follow the instructions as given in **Part 1** on how to configure a router as a bridge. Use the same parameters.
3. On each bridge, delete the contents of the bridge forwarding table with the clear bridge command.
4. Use ifconfig to verify the MAC address (HWaddr) (e.g., eth0 Link encap:Ethernet HWaddr **b6:e4:fb:d2:53:59**) of each PC. Note them down as you will use them to identify the PCs in the bridge forwarding tables with the command show bridge in step 7.
5. Start Wireshark on the Hub to PC1, PC2, PC3 and PC4 links.
6. Clear the ARP cache on all the PCs.
7. Now, issue a set of ping commands. After each command, show the bridge forwarding table on all bridges with the command show bridge, and observe how the ICMP Echo Request and Reply packets travel. Screenshot the bridge forwarding tables.

**PC1%** ping 10.0.1.21 -c 10

**PC2%** ping 10.0.1.31 -c 10

**PC2%** ping 10.0.1.41 -c 10

**PC3%** ping 10.0.1.11 -c 10

1. Stop the traffic capture on each of the 4 links and save the Wireshark output.

### Lab Questions

* Use the captured data to illustrate the algorithm used by bridges to forward packets.
* For **each** of the pings, explain if the learning algorithm resulted in **changes** to the MAC forwarding table. Describe the changes.

### Exercise 2(B). Learning about new locations of hosts

Learning bridges adapt their MAC forwarding tables automatically when the location of a host changes. Due to the learning algorithm, the time it takes to adapt to a change depends on the network traffic and on the value of the *ageing* parameter. This is illustrated in the following exercise.

1. Continue with the configuration of the previous exercise.
2. On each bridge, delete the contents of the bridge forwarding table with the clear bridge command. And clear the ARP cache on each PC with command arp -a.
3. Start Wireshark capture on all 4 PC links.
4. Issue a ping command from PC1 to PC2:

**PC1%** ping 10.0.1.21

1. After a few seconds, disconnect PC2 from Hub2 and connect PC2 to the same hub that PC4 is connected to (Hub4).
2. Record the amount of time that the ping from PC1 to PC2 is not successful after PC2 has been moved to Hub4. (I.e., observe when ICMP packets are successful again in your Wireshark output.)
3. Stop the ping from PC1 to PC2. (^C)
4. Now issue a ping command from PC1 to PC3:

**PC1%** ping 10.0.1.31

1. Then disconnect PC3 from Hub3 and reconnect PC3 to Hub4, the same hub as PC4 is connected to.
2. Immediately generate a set of pings from PC3 to PC1 with the command:

**PC3%** ping 10.0.1.11 -c 5

You will notice that the ping command will fail for a short while and then is successful again.

1. Record the amount of time that the ping from PC1 to PC3 is not successful after PC3 has been moved to Hub4.
2. Stop the ping from PC1 to PC3.
3. Stop the traffic capture on the PC links and save the Wireshark output.

### Lab Questions

* Discuss what you observe in the Wireshark outputs and use that to explain the difference in time that is takes for the pings from PC1 to be successful again in the two ping scenarios above (PC1 to PC2 and PC1 to PC3).

# PART 3. Spanning Tree Protocol

The learning algorithm from Part 3 builds the MAC forwarding tables of bridges, without the need for a routing protocol. However, since learning bridges flood a packet on all ports when a destination is not known, it may happen that packets are forwarded in a cycle and loop indefinitely. The spanning tree protocol for bridges, standardized in the IEEE 802.1d specification, prevents such forwarding loops from occurring. This is done by organizing the bridges in a spanning tree topology. Learning bridges that run the spanning tree protocol are called transparent bridges.

The spanning tree protocol, which is used by virtually all Ethernet switches, works as follows. One bridge, called the ***root bridge***, is elected to be the root of the tree. Each bridge determines which of it ports has the best path to the root bridge. This is the ***root port*** of the bridge. On each LAN, the bridges elect one bridge, called the ***designated*** ***bridge***, which among all bridges on the same LAN, has the best path to the root bridge. The port that connects a bridge to the LAN where it is a designated bridge is called the ***designated port***. Then, all bridges disable (blocking state as opposed to forwarding state) all ports that are not root ports or designated ports. What results is a spanning tree of bridges. Since a tree topology does no have a loop, forwarding packets along the edges of the tree guarantees that forwarding loops are entirely avoided.

This part of the lab has three components: (1) You set up a new network configuration. (2) You verify that bridges without the spanning tree result in forwarding loops. (3) You configure the spanning tree protocol and observe how it prevents loops from occurring. You need to do all parts of this exercise consecutively as you will be using information from each section in another.

### Exercise 3(A). Configuring a topology that results in forwarding loops

1. Setup the network topology as shown in Figure 4.3.

* Routers R1, R2, R3, R4 and R5 are configured as Bridges. Follow the instructions as given in **Part 1** on how to configure a router as a bridge. Use the same parameters.
* PC1 – PC3 IP addresses are given in Table 4.2 below.

**Note:**

For the time being, **do not connect the cables to FastEthernet1/0 of R2 and R4**, shown as dotted lines in Figure 4.3. Making these two connections will result in a forwarding loop. We delay the completion of the loop until you have set up the tools that allow you to observe the forwarding loop.

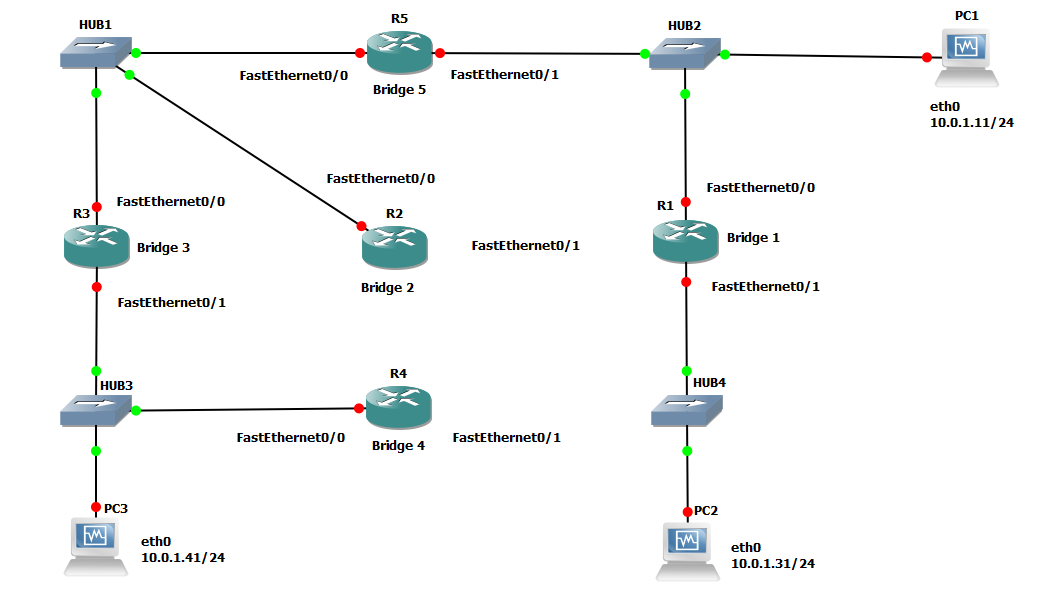


Figure 4.3 Network topology for Part 3.

|  |  |
| --- | --- |
| **PCs** | **eth0** |
| PC1 | 10.0.1.11 / 24 |
| PC2 | 10.0.1.31 / 24 |
| PC3 | 10.0.1.41 / 24 |

Table 4.2 IP Addresses of PCs for Part 3

1. Use ifconfig to verify the MAC address (HWaddr) (e.g. eth0 Link encap:Ethernet HWaddr **b6:e4:fb:d2:53:59**) of each PC. Note them down as you will use them to identify the PCs in the bridge forwarding tables with the command show bridge.
2. **Continue** on to Exercise 3(B).

**Exercise 3(B). Enabling the spanning tree protocol**

In this exercise you will enable the spanning tree protocol on the bridges you setup in Ex 3(A) using the **loop** configuration, i.e., with Bridge2 and Bridge4 **connected** to Hub4. Before starting **3(B)**, we provide a brief description of the spanning tree protocol.

|  |
| --- |
| AN OVERVIEW OF THE SPANNING TREE PROTOCOL  The IEEE 802.1d spanning tree protocol (STP) organizes bridges in a tree topology without any central coordination. Every bridge has only a limited view of the spanning tree, and no bridge has complete knowledge of the spanning tree.  **Bridge ID**: In the spanning tree protocol, each bridge has a unique identifier, called the bridge ID. The bridge ID has a length of 8 bytes. The first 2 bytes are the bridge priority, and the remaining 6 bytes are the bridge MAC address. The bridge priority is a configuration parameter. The bridge MAC address is set to the lowest MAC addressor any of the ports of the bridge. In the spanning tree protocol, the bridge with the lowest bridge ID selected as the root bridge.  **BPDU**: Bridges build the spanning tree by exchanging bridge protocol data units (BPDU). Bridges send BPDUs approximately once every 4 seconds. BPDUs are exchanged only between bridges that are connected to the same LAN.  In Table 4.2 we illustrate the four fields of a BPDU that are relevant to the spanning tree protocol. The BPDU of a bridge advertises the best path from this bridge to the root bridge. Specifically, a BPDU-(R, C, B, P), where R is the value of the root ID, C is the value of the root path cost, B is the bridge ID, and P is the port ID- is interpreted as follows: “I am bridge B and I am sending from my port P. I believe R to be the root bridge, and the cost of my path to the root bridge is C.” |

| Length (in Bytes) | Field Name | Content |
| --- | --- | --- |
| 8 | Root ID | Identifies the root bridge (as seen by the sender of this BPDU) |
| 4 | Root path cost | Cost of the path from the sender of this BPDU to the root bridge |
| 8 | Bridge ID | Identifies the sender of this BPDU |
| 2 | Port ID | Identifies the network interface (port) where this BPDU is sent. |

Table 4.2. Fields of a BPDU relevant to the spanning tree construction.

|  |
| --- |
| **Operation of the spanning tree protocol**  Each bridge listens on all its ports to BPDUs sent by other bridges. If a bridge receives a BPDU that advertises a “better” path than advertised in its own BPDU, the bridge updates its BPDU. To determine if a received BPDU advertises a better path, the bridge compares the received BPDU to its own BPDU. If the root ID in the received BPDU is smaller than the root ID of the bridge, the received BPDU is seen as advertising a better path. If the root ID are identical, the BPDU with the lower root path cost advertises a better path. If both the root ID and root path cost are identical, then the BPDU with the lower bridge ID is seen as advertising a better path. Finally, if root ID, root path cost, and bridge ID are all identical, then the BPDU with the lowest port ID is interpreted as advertising a better BPDU.  When a bridge with BPDU (R1, C1, B1, P1) receives a BPDU (R2, C2, B2, P2) and the received BPDU advertises a better path, the bridge updates its own BPDU to (R2, C2+increment, B1, P1). The increment value is a configuration parameter that accounts for the cost increase of the path due to bridge B1 being now on the path. When the increment value is set to 1 on all bridges, then the bridges establish a minimum hop route to the root bridge. The increment can also be set to account for the data rate of a LAN. For example, to make a path on a 100 Mbps LAN more desirable than on a 10 Mbps LAN, the 10 Mbps LAN can be assigned a larger increment value.  A bridge transmits its BPDU on a port only if its BPDU advertises a better route than any of the BPDUs received on that port. In this case, the bridge assumes that it is the designated bridge for the LAN to which the port connects, and the port that connects the bridge to the LAN is called the designated port of the LAN. A bridge that is not the designated bridge for a LAN does not send BPDUs on that LAN.  **Constructing the spanning tree**  Each bridge locally decides which of its ports are part of the spanning tree. Only the root port and the designated ports of a bridge are part of the spanning tree; the other ports are not part of the spanning tree. One can reconstruct the complete tree by connecting, for each LAN, the root ports that connect to this LAN with the designated port of the LAN.  Each bridge forwards packets only on ports that are part of the spanning tree, that is, if they are received on the root port or sent on its designated ports. These ports are said to be in a ***Forwarding*** state. All other ports are said to be in a ***Blocking*** state. In this way, packets are forwarded only along the edges of the spanning tree. As a result, since a tree topology does not have a loop, the forwarding of packets does not result in loops.  **Initializing the spanning tree protocol**  When a bridge, say, with bridge ID B, is started, it assumes that it is the root bridge. It sends a BPDU (B, O, B, p) on all its ports p. The root path cost is set to 0, since B believes itself to be root. Within a short amount of time, the bridge learns about better paths, and the protocol quickly on verges to a new spanning tree. |

Your task in this exercise is to capture the BPDU packets and use the show spanning-tree command to analyze how the spanning tree is built.

1. Complete the topology in Figure 4.3. by connecting interface FastEthernet1/0 of Bridge2 and Bridge4 to Hub4 as shown in Figure 4.3. Setup the 1/0 interface on each of Bridge 2 and Bridge4.
2. Now set-up the bridges to “enable” the Spanning Tree Algorithm (SPT). Commands for Bridge1 are shown below. Repeat for each bridge.

**R1#** configure terminal

**R1(config)#** interface FastEthernet0/0

**R1(config-if)#** bridge-group 1

**R1(config-if)#** no bridge-group 1 spanning-disabled

**R1(config-if)#** interface FastEthernet0/1

**R1(config-if)#** bridge-group 1

**R1(config-if)#** no bridge-group 1 spanning-disabled

**R1(config-if)#** exit

1. Wait for a minute until the spanning tree stabilizes. The spanning tree has stabilized when all interfaces of the bridges are no longer in “**Learning**” state, but are either in the “**Blocking**” or “**Forwarding**” state. To check on the state of a bridge, use the show spanning-tree command as shown below for bridge R1:

**R1#** show spanning-tree

1. Start Wireshark capture on all 3 PC links. To simplify your analysis, filter for spanning tree BPDUs, ARP and ICMP messages - e.g. enter “arp || icmp || stp” as the display filter.
2. Issue a ping command from PC1 (10.0.1.11) to PC3 (10.0.1.41).

**PC1%** ping 10.0.1.41 -c 10

1. The ping will be successful if all is configured correctly and the spanning tree is stable.
2. On each bridge use the show spanning-tree command. Save the output of the show spanning-tree by taking a screen shot.
3. Use the show bridge command on each bridge to show the forwarding table. Screen shot and save the output.
4. Use the live Wireshark data (stp BPDUs) to determine the relevant information of the spanning tree, such as the root bridge, the root ports of each bridge, the designated bridges, the designated ports, the forwarding ports, and the blocked ports.
5. Stop Wireshark capture and save the output.
6. Do not STOP GNS3. Please check **NOTE** below.

### Lab Questions

* + Using the show spanning-tree output, draw the spanning tree as seen by the bridges. Include all relevant information, such as a root bridge, root ports, designated bridges, designated ports, and blocked ports.
  + Identify which bridges were sending the stp BPDUs on each segment, and check the following fields for each such BPDU: root ID, root path cost, and the bridge ID. Explain how the messages are interpreted. Show how the entire spanning tree can be constructed from these messages (without the output of the show spanning-tree command).
  + What is the default path cost for each hop between bridges?
  + Use Figure 4.3 to trace the packets resulting from the ping command (e.g. ARP Request and Reply, ICMP Echo Request and Reply). Justify your answer with the saved data.
  + Use the forwarding tables of the bridges and the saved MAC address of the PCs to trace the path taken by the ICMP packets.

|  |
| --- |
| **Note:**  For Part 4, you need to remember which bridge was the **root bridge.** If you don’t do Part 4 immediately, you will have to rerun Exercise 3(B) to determine the root bridge before proceeding with the exercises in Part 4. |

# Part 4. Adaptation of the Spanning Tree to Network Changes

The spanning tree protocol adapts to changes in the network topology. Here you observe how the spanning tree topology adapts when a link fails.

### Exercise 4(A). Reconfiguration of the spanning tree

You will simulate a failed link in the network shown in Figure 4.3 by disconnecting an Ethernet cable, i.e., **shutting down the interface**. You will observe how the bridges adjust to a change in the network topology.

1. Use the network configuration as shown in Figure 4.3 with all links connected as in Ex 3(B).

Make sure that the spanning tree protocol is enabled on Bridges 1-5 and that the priorities of all bridges are set to 128.

1. Run Wireshark on the links connected to PC1, PC2, and PC3.
2. From PC1 (10.0.1.11) issue 100 pings to PC3 (10.0.1.41)

**PC1%** ping 10.0.1.41 -c 100

1. Disconnect one cable connected to (i.e., **shutdown one interface** of) the **root bridge** (recalled from Ex 3(B)). If you did not just complete Exercise 3(B), you will need to determine the root bridge by completing Exercise 3(B) again.
   * BPDUs will be used to build a new spanning tree. Observe that the ping command at PC1 will be unsuccessful for some time, but will return to its successful state when a new spanning tree is built.
   * Measure the time it takes to build a new spanning tree using the Wireshark data.
2. Use the show spanning-tree command on each bridge. Screen shot and save.
3. On each bridge use the show bridge command to display the forwarding table. Screenshot and save.
4. Stop the Wireshark capture. Continue to Ex4(B) after answering the question below.

### Lab Questions

* Using the saved output of show spanning-tree, draw the spanning tree as seen by the bridges. For each bridge indicate: the root ports, designated ports, and blocked ports. Briefly explain the changes in the spanning tree.
* Using the forwarding table of the bridges, trace the path that the ARP and the ICMP packets take (use the saves PC MAC addresses to identify the route).

### Exercise 4(B). Configuring a bridge to be the root bridge

In this exercise, you force a certain bridge to become the root bridge. Recall that the priority of a bridge is used in the first 2 bytes of the bridge ID. Thus, the bridge with the lowest priority value has the lowest bridge ID. Since the spanning tree protocol elects the bridge with the lowest bridge ID as the root bridge, the root bridge can be fixed by modifying the priority field.

Being able to force a certain bridge to be the root bridge provides some control over the spanning tree topology. For example, one can select the device with the highest capacity (most ports), to become the root bridge, or the most centrally located bridge, etc..

1. Still using the configuration with one cable disconnected on the root bridge from Ex 4(A), choose any **non-root** bridge to become the new root bridge here.
2. Change the priority of this **non-root** bridge to 64 to force it to be the new root. (Note that all the other bridges will still have a bridge priority of 128.) To change the priority of a bridge, use the following commands:

**R#** configure terminal

**R(config)#** bridge 1 priority 64

**R(config)#** end

1. Start Wireshark on all 3 PC links.
2. From PC1 (10.0.1.11) issue 100 pings to PC3 (10.0.1.41)

**PC1%** ping 10.0.1.41 -c 100

1. BPDUs will be used to build a new spanning tree. Observe that the ping command at PC1 will be unsuccessful for some time, but will return to its successful state when a new spanning tree is built.
2. Wait until the tree stabilizes. Then use the show spanning-tree command on each bridge. Screenshot and save.
3. From the output, verify that the bridge you assigned priority “64”, has been selected as the root.
4. On each bridge use the show bridge command to display the forwarding table. Screenshot and save.
5. Stop the Wireshark capture. Stop GNS3.

### Lab Questions

* + Draw the new spanning tree as seen by the bridges.
  + Trace the path that the ARP and the ICMP packets take through the bridges using the forwarding tables and the PC MAC addresses.Is it different from part 4(A) above? Why?

# Part 5. Mixed Router and Bridge Network Topology

In this part of the lab, you set up a network topology that contains bridges as well as IP routers. Both bridges and routers are devices that connect networks and forward packets between networks. Bridges make forwarding decisions based on destination MAC addresses. IP routers make forwarding decisions based on destination IP addresses. In a properly configured network, bridges and IP routers coexist without causing network problems. Sometimes, however, the forwarding of packets in a network with bridges and IP routers can be difficult to trace. The following exercises explore such a scenario. Note the different subnets and how they are interconnected. It will test your knowledge of subnets, Proxy ARP and more.

### Exercise 5(A). Setting up the network configuration

1. The IP configuration of all 4 PCs and routers R2 and R3, are given in Tables 4.3 and 4.4.

**NOTE**: The network prefix of PC4 has been changed to /**16** as shown in Table 4.3. All other IP network interfaces have /24 netmasks.

A screenshot of a video game

Description automatically generated

Figure 4.4. Network topology for Part 5

| Cisco Routers | eth0 | Default Gateway |
| --- | --- | --- |
| PC1 | 10.0.1.11 / 24 | 10.0.1.2 |
| PC2 | 10.0.3.21 / 24 | 10.0.3.2 |
| PC3 | 10.0.4.31 / 24 | 10.0.4.3 |
| PC4 | 10.0.4.41 / **16** | 10.0.4.3 |

Table 4.3. PC configurations for Part 5

| Cisco Routers | Configured As | Configuration Information |
| --- | --- | --- |
| R1  R4 | Bridge | * Enable bridging on both interfaces FastEthernet0/0 and FastEthernet1/0 * The spanning tree protocol can be enabled or disabled. * IP addresses need not be configured. |
| R2 | IP Router | * IP configuration on FastEthernet0/0: 10.0.3.2/24 * IP configuration on FastEthernet1/0: 10.0.1.2/24 * Default gateway set to 10.0.3.3 * IP forwarding is enabled. |
| R3 | IP Router | * IP configuration on FastEthernet0/0: 10.0.3.3/24 * IP configuration on FastEthernet1/0: 10.0.4.3/24 * Default gateway set to 10.0.3.2 * IP forwarding is enabled. |

Table 4.4. Routers and Bridges configurations for Part 5

1. Connect the PCs, Bridges, and Routers as shown in Figure 4.4.
2. Configure the PCs with the IP addresses and default gateways as given in Table 4.3.
3. R1 and R4 as bridges. Configure as shown before in Part 1.
4. In this exercise, R2 and R3 ROUTERS.

The configuration commands for router R2 (repeat for R3 using the appropriate IP addresses as given in Table 4.4): (note that we enable IP forwarding again).

**R2#** configure terminal

**R2(config)#** ip routing

**R2(config)#** ip route 0.0.0.0 0.0.0.0 10.0.3.3

**R2(config)#** interface FastEthernet0/0

**R2(config-if)#** ip address 10.0.3.2 255.255.255.0

**R2(config-if)#** no shutdown

**R2(config-if)#** interface FastEthernet1/0

**R2(config-if)#** ip address 10.0.1.2 255.255.255.0

**R2(config-if)#** no shutdown

**R2(config)#** end

1. Setup the default gateways for R1 and R2 as given in Table 4.4.

### Exercise 5(B). Observing traffic flow in a network with IP routers and bridges

Here you observe the paths of packets between the PCs. You will see that in a mixed IP router and bridge environment, tracing the path of a packet is not always straightforward.

1. Clear the forwarding tables on all bridges and clear the ARP table on all PCs and Routers.
2. Run Wireshark on all PC links.
3. Issue the following ping commands. Save the traffic captured by Wireshark and examine the outputs to determine the path of the ICMP Request and Reply packets.

**PC1%** ping 10.0.4.31 -c 10

**PC1%** ping 10.0.4.41 -c 10

**PC4%** ping 10.0.1.11 -c 10

**PC1%** ping 10.0.3.21 -c 10

**PC4%** ping 10.0.3.21 -c 10

1. Stop the traffic capture on all links. For each Wireshark output, select and save enough ICMP packets so that you can address the questions raised in below.
2. Stop GNS2. Quit GNS3.

### Lab Questions

* Determine which of the ping commands are successful and which fail.
* Use the data displayed by Wireshark to determine the route of the ICMP Echo Request and Reply packets (e.g. PC1 -> Bridge 1 -> R2 -> Bridge 2 -> PC4) for each ping.
* For each path, provide an explanation why a certain route is taken by the ICMP Echo Request and Reply packets.