

Networking Lab Day 6

Kevin Neumann, Alain C. Mendy

June 9, 2014, Goettingen

1 Prelab

The prelab preparations can be found in the directory prelab in the known git repository (<https://www.github.com/Phiology/networkingLab>).

2 Part 1: Configuring a Linux PC as a bridge, observing a bridge in operation and manipulating a PC bridge

2.1 Does the bridge manipulate any of the fields in the MAC and IP headers?

No, it does not.

2.2 Do the source and destination MAC/IP addresses change when a packet traverses a bridge? Provide an explanation and include an example from the captured data. Suppose that PC2 was configured as an IP router, which differences would you observe in the Ethernet and IP headers?

No, the source and destination MAC and IP addresses do not change, because a bridge is transparent in the network. If PC2 was configured as a router, the IP address would not change, but the MAC addresses would. First packet goes from MAC-PC1 to MAC-PC2 and second from MAC-PC2 to MAC-PC3.

2.3 Include the output of the traceroute command from Step 5. Provide an explanation why PC2 does not appear in the output of the traceroute command in Step 6. Include the answers to the questions in Step 6.

```
traceroute to 10.0.1.31 (10.0.1.31), 30 hops max, 38 byte packets 1 10.0.1.31  
(10.0.1.31) 0.844 ms 0.680 ms 0.713 ms
```

PC2 does not appear in traceroute, because it is configured as a bridge and thus is transparent in the network. If PC2 was configured as a router, the hop would be seen in the traceroute output.

2.4 Does the ping command from PC1 to PC3 still work?

Yes it does, because it can pass the bridge and still find PC3.

3 Part 2: Configuring a Cisco router as a bridge

3.1 Compare the results to the outcome of the traceroute command in Exercise 1C.

The traceroute output is identical.

3.2 Why is it not possible to issue a ping command to Router1?

Because it is configured as a bridge and thus operates on a MAC-Address level and is transparent in the network.

4 Part 4 : Exploring the learning algorithm of bridges, and learning about new locations of hosts.

4.1 Use the captured data to illustrate the algorithm used by bridges to forward packets.

Address	Action	Interface	Age	RX count	TX count
0000.c058.2800	forward	FastEthernet0/0	3	3	2
0004.75ad.04ab	forward	FastEthernet0/0	4	1	0
0050.bf75.209a	forward	FastEthernet0/0	4	3	3
0050.bf73.5aab	forward	FastEthernet0/1	3	6	5

When a packet arrives, the bridge tries to find the outgoing port to the destination MAC-address if known. If it isn't in the forwarding table, it floods the package on all ports except for the one where the packet arrived.

4.2 For each of the transmitted packets, explain if the learning algorithm results in changes to the MAC forwarding table. Describe the changes.

After the first ping, the bridges Router1 and Router2 know the ports to PC1 and PC2 and the bridge on Router3 knows PC1. After the second ping there are no changes but the TX and RX fields increase. After the third ping from PC2 to PC4, all bridges know the ports for PC4. The last ping from PC3 to PC2, all the bridges know PC3, thus completing the bridge forwarding tables.

4.3 Explain this outcome and compare it to the outcome of Step2.

The ping from PC1 to PC3 first failed, but it became successful after the bridges know about PC3's new location once a ping was issued from PC3.

5 Part 7: Observing traffic flow in a network with IP routers and bridges

- 5.1 Describe which of the ping commands are successful and which fail. Use the data that you captured to determine the route of the ICMP Echo Request and Reply packets. For each route, provide an explanation why the path is taken for each of the ping commands.

Failed: PC1 to PC3. The Reason for that is that PC3 is in a different subnet.

Succeeded:

PC1 to PC4 with the route PC1 – > bridge1 – > router2 – > router 3 – > PC4. The path is chosen because router2 is the default gateway of PC1.

PC4 to PC1 with the exact reverse of the above mentioned route.

PC1 to PC2 via bridge1 – > router2 – > PC2. Both PC's default gateways is router2.