Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Spring Term 2016
Prof. Adrian Perrig

Operating Systems and Networks
Solution 8

Note: these solutions constitute material *supplemental* to the exercise sessions.

1 ARP: Hands-on

In this exercise you will use Wireshark to examine the ARP protocol in more detail. You will ping the IP address 8.8.8.8 and analyze the exchanged messages.

(1) MAC addresses are stored in an ARP cache to avoid successive ARP requests. In order to ensure that ARP messages will be exchanged, you have to clear the ARP cache. Use the man page of the `arp` command to find out how to delete ARP cache entries. Verify that the table is empty using e.g., `arp -nv` on Linux. What is the approximate lifetime of an ARP cache entry?

Answer: ARP cache entries have a typical lifetime of 20 minutes, after which the entry is marked as stale and is refreshed. The behaviour varies according to the operating system, but 5-20 minutes should be enough for all OSes. Arp entries can be deleted via `arp -d mac-address` depending on the OS (this example is for Linux).

(2) Open Wireshark and start capturing incoming and outgoing traffic. Use the `ping` command to send ICMP echo requests to the IP address 8.8.8.8. Locate the ARP messages exchanged (you can use appropriate display filters) and note the Ethernet and IP source and destination addresses. To which entities do these addresses belong and why? **Hint:** Use the command `ip route show` on the Linux command line to check the local routing table. After running ping, check the ARP table entries with the `arp` command. Use the results to help explain you answer.

Answer The only destination MAC address that should appear in the table should be that of the default gateway (default router). The source MAC address should belong to the system issuing pings. Since 8.8.8.8 is outside the local network, no ARP will be attempted for this host.

2 Traceroute: Hands-on

(1) Use the traceroute utility (tracert on Windows) to trace the route from your system to the following universities in other countries. For each traceroute output, find the transoceanic link(s) noting the 2 endpoints' (of the longest hop) IP addresses and the delay between them.

a) www.mit.edu (Boston, USA)

b) www.ubc.ca (Vancouver, Canada)

c) www.u-tokyo.ac.jp (Tokyo, Japan)

d) ucl.ac.uk (London, UK)

3 ICMP

ICMP is usually listed as a layer 3 (network layer) protocol. As data gets sent to lower layers, each underlying layer adds its own header. Note, however, that ICMP messages use an IP (another layer 3 protocol) header to encapsulate their data. Could ICMP be better classified as a layer 4 (transport) protocol? Justify your answer. Hint: the beginning of RFC 792 (`https://tools.ietf.org/html/rfc792`) may provide clues.

Answer: The answer depends on your interpretation of OSI model layers. ICMP is not designed as a standalone layer 3 protocol, but rather as an extension to another one (IP). One valid answer would be to say that ICMP is layer 4 because a layer 3 header is added to these packets. Some might argue that it is 3 because it must be implemented alongside IP. Some might say it is 3.5 or "at the top" of layer 3.

4 NAT and Private Address Space

RFC 1918 "Address Allocation for Private Internets" (`https://tools.ietf.org/html/rfc1918`) specifies 10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16 as private addresses suitable for unrestricted private internal use. Many home networks use the 192.168.1.0/24 address space and use NAT to a single public address.

(1) Could 172.28.2.35/24 be used as the address space for an internal network? Explain your answer.

Answer: Yes. 172.28.x.x falls under the /12. The range (from the RFC) is 172.16.0.0-172.31.255.255

(2) Could 10.255.255.0/24 be used as the address space for an internal network? Explain your answer.

Answer: Yes. that range is within the private space. No problem by using 255 in intermediate octets.

(3) Assume a technically illiterate user configures 8.8.8.0/24 as the internal address space on his/her home router. Assume that NAT is used at the router, and there is a public interface on that router with IP address 19.33.93.140. Briefly explain what happens if a user at a system with IP 8.8.8.2 attempts to establish a network connection with a system on the Internet at IP address 8.8.8.8 (Google public DNS). Can the two systems communicate? Why? Why not?

Answer: Any internal machine trying to communicate with a computer outside the network with IP 8.8.8.8 would assume that 8.8.8.8 is on the same local segment, thus would attempt to find the destination on the local network rather than doing NAT and forwarding to the Internet.

5 Longest Prefix Match 1

The following is a forwarding table at router R. Suppose packets with the following destination IP addresses arrive at router R. Determine the next hop. Give only one answer for each destination, using longest prefix matching.

Destination Network	Next Hop
139.179.200.0/23	R1
139.179.128.0/18	R2
139.179.120.0/20	R3
139.179.220.0/21	R4
139.179.0.0/16	R5

(a) 139.179.60.10 (b) 139.179.226.40 (c) 139.179.124.55 (d) 139.179.220.180

Answer:

Let us write down the networks in binary notation.

139.179.200.0/23	10001011.10110011.11001000.x	R1
139.179.128.0/18	10001011.10110011.10000000.x	R2
139.179.120.0/20	10001011.10110011.01111000.x	R3
139.179.220.0/21	10001011.10110011.11011100.x	R4
139.179.0.0/16	10001011.10110011.00000000.x	R5

The ranges are thus

10001011.10110011.11001000.x	R1	from 139.179.200.0	to 139.179.201.255
10001011.10110011.10000000.x	R2	from 139.179.128.0	to 139.179.191.255
10001011.10110011.01111000.x	R3	from 139.179.112.0	to 139.179.127.255
10001011.10110011.11011100.x	R4	from 139.179.216.0	to 139.179.223.255
10001011.10110011.00000000.x	R5	from 139.179.0.0	to 139.179.255.255

We thus get the following next hops:

(a) R5 (b) R5 (c) R3 (d) R4

6 Longest Prefix Match 2

Without using longest prefix matching, a forwarding table looks like the one below on the left. If we use longest prefix matching, we can combine a few entries. Fill in the reduced table on the right.

Answer: Only one row can be removed from the table.

First thing to notice is that the outgoing interface must be the same for all rows that are possibly merged. We thus only consider the three rows with `eth0`. The binary notation of these `eth0` rows looks as follows.

128.128.0.0/9	1000000.1000000.x.x	eth0
128.160.0.0/11	1000000.1010000.x.x	eth1
128.176.0.0/12	1000000.1011000.x.x	eth0
128.192.0.0/10	1000000.1100000.x.x	eth0

In other words, whatever goes to `10000000.1x.x.x` should be routed via `eth0`, with the one exception of `10000000.101.x.x` via `eth1`. Since the prefix length is 11 in this case, one can merge the /9 and /10 rules for `eth0`, since their prefix is less than 11. One hence obtains

128.128.0.0/9	1000000.1000000.x.x	eth0
128.160.0.0/11	1000000.1010000.x.x	eth1
128.176.0.0/12	1000000.1011000.x.x	eth0

which corresponds to the following in decimal notation.

Prefix	Outgoing Interface
128.128.0.0/9	eth0
128.160.0.0/11	eth1
128.176.0.0/12	eth0
128.192.0.0/10	eth0
default	eth2

Prefix	Outgoing Interface
128.128.0.0/9	eth0
128.160.0.0/11	eth1
128.176.0.0/12	eth0
default	eth2

If we are allowed to change existing rules, then a more compact solution is the following.

Prefix	Outgoing Interface
128.128.0.0/9	eth0
128.160.0.0/12	eth1
default	eth2