

Computer Networks Theory + Lab

CS252 Labs ▾ > Lab07: IP Packet Format, DHCP and ICMP ▾ ▾

Summary

Contents

1. [Lab07: Instructions \(Document\)](#)

[« Previous \(Lab06: IP Forwarding and ARP\)](#)[Next \(Lab08: Sockets and TCP\) »](#)

Lab07: Instructions

Lab07: Exploring IP Packet format, DHCP and ICMP protocols

Objective:

1. Dig deeper into the following network layer concepts: packet formats, DHCP and ICMP

(Watching the video “Digging deeper into Network Layer” is a pre-requisite for this lab.)

Reference Material:

1. Digging deeper into Network Layer (both parts)
2. IP packet format
3. IPv6
4. DHCP
5. ICMP

Requirements:

1. Linux Command Terminal
2. Wireshark

Exercise 1: Packet Format

Part-A: IP Fragmentation

We have studied about IP fragmentation. Let us see it in action. We will experiment with two cases:

- 1) When no fragmentation occurs and 2) When fragmentation occurs.

Guidance:

1. Download all relevant files related to this lab by clicking on “View Document” at the bottom. Unzip it via “unzip student-files.zip”.
2. “sendUDP.c” is a simple socket program (provided in the directory) that generates a single IP packet of a given size and sends it to the specified destination IP address. Compile the program and generate the executable via “gcc sendUDP.c -o sendUDP”.
3. Run wireshark and while it is running, execute the code via “./sendUDP remote-host pkt-size”. Example “./sendUDP 10.129.5.6 300”. This command send a packet of size 300 bytes to host 10.129.5.6.
4. Experiment with different packet sizes, but to avoid confusion, you may want to capture each run of sendUDP in a separate wireshark trace. Use filters based on “ip.addr” to identify the packets belonging to a given experiment.
5. It is strongly recommended that you do this experiment yourself and collect your own traces. We have however provided a single trace (packet-format.pcapng) that corresponds to both the cases (to make life a bit more interesting :-), in case you have not been able to collect such traces.
6. You should also know the significance of fragmentation fields (look at slides 11-16 of IP-protocol-slides.pdf on BodhiTree)

Now attempt the first module of the quiz titled “Lab07-Packet-Format”.

Part-B: IPv6

Ipv6 header differs from IPV4. Let us explore this difference by looking at the respective header formats (see slide 4 of ipv6-slides.pdf on BodhiTree). You can collect your own trace by pinging an Ipv4 and an IPv6 address. We have however provided a single trace (ipv4-ipv6.pcap) in case you have not been able to collect.

Now attempt the second module of the quiz titled “Lab07--Packet-Format”.

Exercise 2: Dynamic Host Configuration Protocol (DHCP)

Configuring IP addresses requires root permission. Further most sysads in a computer lab would not want you to tinker with the IP addresses of the lab machines. You are most welcome to generate a DHCP trace on your laptop as outlined in the video corresponding to “Digging deeper into the Network Layer part—2” and explore it. But for this exercise use the provided trace “dhcp.pcap”, which is the result of executing the below three commands in that order:

1. “dhclient -v eno1”
2. “dhclient -v -r eno1”
3. “dhclient -v eno1”

Now attempt the quiz titled “Lab07-DHCP”.

Exercise 3: Internet Control Message Protocol (ICMP)

Part A: Traceroute/mtr

When a flow between two end points is not performing as per expectation, it is useful to debug what is happening along the path. Is it too many routers, is it the delay etc?

Guidance:

1. For this exercise, you should try both `tracert` and `tracert -n` commands. `Tracert` is more easy to check. The `tracert -n` command is a combination of `ping` and `tracert` commands.

Part-B: IPv6

Ipv6 header differs from IPV4. Let us explore this difference by looking at the respective header formats (see slide 4 of [ipv6-slides.pdf](#) on BodhiTree). You can collect your own trace by pinging an Ipv4 and an IPv6 address. We have however provided a single trace ([ipv4-ipv6.pcap](#)) in case you have not been able to collect.

Now attempt the second module of the quiz titled “Lab07–Packet-Format”.

Exercise 2: Dynamic Host Configuration Protocol (DHCP)

Configuring IP addresses requires root permission. Further most sysads in a computer lab would not want you to tinker with the IP addresses of the lab machines. You are most welcome to generate a DHCP trace on your laptop as outlined in the video corresponding to “Digging deeper into the Network Layer part—2” and explore it. But for this exercise use the provided trace “[dhcp.pcap](#)”, which is the result of executing the below three commands in that order:

1. “`dhclient -v eno1`”

Part-B: IPv6

Ipv6 header differs from IPV4. Let us explore this difference by looking at the respective header formats (see slide 4 of [ipv6-slides.pdf](#) on BodhiTree). You can collect your own trace by pinging an Ipv4 and an IPv6 address. We have however provided a single trace ([ipv4-ipv6.pcap](#)) in case you have not been able to collect.

Now attempt the second module of the quiz titled “Lab07–Packet-Format”.

Exercise 2: Dynamic Host Configuration Protocol (DHCP)

Configuring IP addresses requires root permission. Further most sysads in a computer lab would not want you to tinker with the IP addresses of the lab machines. You are most welcome to generate a DHCP trace on your laptop as outlined in the video corresponding to “Digging deeper into the Network Layer part—2” and explore it. But for this exercise use the provided trace “[dhcp.pcap](#)”, which is the result of executing the below three commands in that order:

1. “`dhclient -v eno1`”
2. “`dhclient -v -r eno1`”
3. “`dhclient -v eno1`”

Now attempt the quiz titled “Lab07-DHCP”.

Exercise 3: Internet Control Message Protocol (ICMP)

Part A: Traceroute/mtr

When a flow between two end points is not performing as per expectation, it is useful to debug what is happening along the path. Is it too many routers, is it the delay etc?

Guidance:

1. For this exercise, you should try both `tracert` and `mtr` commands. `Tracert` is more easy to check. The `mtr` command is a combination of `ping` and `tracert` commands.